卒業製作 2003 年度 (平成 15 年度)

終端ネットワークにおける
複数ルータの利用に関する研究

指導教員

徳田 英幸

村井 純

楠本 博之

中村 修

南 政樹

慶應義塾大学 環境情報学部

清水 崇史

t00473ts@sfc.keio.ac.jp

平成 16 年 1 月 15 日

卒業論文要旨 2003 年度 (平成 15 年度)

終端ネットワークにおける複数ルータの利用に関する研究

本研究では、終端ネットワークにおける経路制御の問題点を明らかにし、その解決方法を示した。

現在、世界中に広まっているインターネットは、その基幹部で動的経路制御プロトコルを用い、経路の冗長化を図っている。しかし、インターネットの末端に位置する終端ネットワークでは、動的経路制御プロトコルを用いることが一般的ではない。そのため、終端ネットワークの出口となるルータに障害が発生した場合に代替経路への経路変更を行う事が出来ない。このように、一地点の障害によって、ネットワーク全体の接続性が失われることをシングルポイント障害と呼ぶ。経路の冗長化を図る事が困難な終端ネットワークにおいて重大な問題となっている。

本研究では、この問題を解決するために、VRRP(Virtual Router Redundancy Protocol) を利用した。VRRP は動的経路制御が不可能な環境において、静的に設定された経路の冗長化を図るためのプロトコルである。本研究では、現状における VRRP の問題点を明らかにした。そして、次世代のインターネットシステムである IPv6 に対応した VRRP をルーティングパッケージである Zebra 上に実装し、評価を行った。

本研究の結果、終端ネットワークにおける複数ルータによる経路冗長化を実現した。

キーワード

1: 終端ネットワーク　2: 複数ゲートウェイ　3: VRRP

慶應義塾大学 環境情報学部
清水 崇史

Abstract of Bachelor's Thesis Academic Year 2003

Improving router redundancy in end-user network

In this research, problems on routing at end network is studied and a solution for such problems is proposed.

The Internet uses dynamic routing protocols on its backbone network to provide routing redundancy. However, dynamic routing protocols are not used at most of the end networks. Such implementation of the network result in the situation where end nodes lose network connectivity when the gateway of end network fails. This is due to the fact that end nodes does not have any method to change its routing table dynamically. "Single point of failure" of end network is a phrase used to describe the situation mentioned above. It becomes critical problem to the end network that has difficulties in providing routing redundancy

VRRP(Virtual Router Redundancy Protocol) is used to approach "single point of failure" of end network. An idea of VRRP is a protocol to provide redundancy to static routing in the environment where dynamic routing is not available.

In this research, problem of VRRP is clarified and proposed a revision of its specifications. Also IPv6 capable VRRP is implemented on the routing package called Zebra and then evaluated.

As a result, providing route redundancy using multiple gateways for end networks is realized.

Keywords

1: End-user Network 2: Multi Gateways 3: VRRP

Faculty of Environmental Information, Keio University
Takashi Shimizu

目 次

図 目 次

表 目 次

第1章　序論

本章では、まず本研究の前提となる背景について述べ、次に本研究の目的を述べる。最後に本論文の構成を述べる。

1.1　本研究の背景

インターネットは、世界中に普及したネットワークであり、学術機関やビジネス等で広く使われている。現在インターネットでは、データの転送に IPv4 (Internet Protocol version 4) [1] が用いられている。しかし、IPv4 の 32 ビットのアドレス空間が枯渇し、それが深刻な問題になっている。それに代わる次世代のプロトコルとして、128 ビットのアドレス空間を持つ IPv6 (Internet Protocol version 6)[2] の開発が進んでいる。

インターネットはデータの転送に、ルータ [3] と呼ばれる経路制御を行う特殊なコンピュータを利用する。各ルータは IP パケットと呼ばれる単位のデータを目的のノードまで、経路表 (ルーティングテーブル) を基にバケツリレー方式で転送する。各ルータの経路表を、トポロジの変更に従い動的に制御するのが動的経路制御プロトコルである。

また、インターネットは階層的に構成されている。最も大きい単位のネットワークは AS (Autonomous System) と呼ばれる。AS の例には ISP (Internet Service Provider) や学術ネットワークがある。AS 内には数多くの小さいネットワークが存在する。それぞれのネットワークでは、その規模に対応した動的経路制御プロトコルが動作し、経路の安定性を高めている。

このような階層的なインターネットの中で、末端に位置する終端ネットワークでは、動的経路制御プロトコルを利用することは一般的ではない。ユーザの利用する終端ノードは、デフォルトゲートウェイと呼ばれる第一ホップルータを静的に指定する。そのため、デフォルトゲートウェイに障害が発生した場合、別の経路への切り替えを行う事ができない。よって、終端ノードはインターネットへの接続を断たれてしまう。このように、終端ネットワークは基幹ネットワークに比べ、経路障害への対策が不十分である。

インターネットは終端ノードのためのネットワークである。デフォルトゲートウェイのシングルポイント障害は非常に重要な問題である。

1.2　本研究の目的

本研究では、終端ネットワークにおける第一ホップルータのシングルポイント障害の問題を解消することを目的とする。

終端ネットワークのデフォルトゲートウェイの冗長化を目的とした機構に VRRP (Virtual Router Redundant Protocol)[4] がある。しかし、シングルポイント障害を解消するためには、ただ VRRP を利用するだけでは不十分である。経路制御プロトコルや RA (Router Advertisement)[5] など、上下流の経路制御機構との連携を考えなくてはならない。

本研究では、上下流の経路制御システムとの連携を考えたシングルポイント障害問題の解決方法を示す。また、ルーティングパッケージである Zebra に VRRP を実装し、パッケージ内の他のプロトコルとの連携を可能とする運用方法を示す。これにより、終端ネットワーク環境の信頼性を向上させる。

1.3　本論文の構成

本論文では、第２章にて、本研究が取り扱う範囲である終端ネットワークを整理し、その分析を行う。第３章では、VRRP の詳細を述べた。第４章では、終端ネットワークの問題点をまとめ、それに対する本研究のアプローチを示した。第５章では、本システムの設計を述べた。第６章では、本システムの実装に関する詳細を述べた。最後に第７章で、本研究の評価を行った。

第2章　現状の終端ネットワーク

本章では、本研究における背景となるインターネットの現状を述べる。

2.1　インターネットにおける階層的経路制御の発達

インターネットは、各地に散在する数多くの独立したコンピュータネットワーク同士を接続した「ネットワークのネットワーク」である。1990 年代の初頭より商用での利用が始まり、メールや WWW 技術の開発などから、現在は世界中に普及しているネットワークに成長した。

現在利用されているインターネットはデータの転送に IPv4 (Internet Protocol version 4) [1] というプロトコルが用いられている。ルータと呼ばれる経路制御を行う特殊なコンピュータを利用し、IP パケットを単位とするデータを目的のノードまで転送する。各ルータは、経路表 (ルーティングテーブル) を基に、IP パケットをバケツリレー方式で転送する。経路表の作成方法には静的ルーティングと動的ルーティングの二種類がある。静的ルーティングは管理者が各ルータの経路表を手動で設定する方法である。逆に、経路制御プロトコル (ルーティングプロトコル) を用いて各ルータの経路表を自動的に作成するのが動的ルーティングである。現在、IPv4 の 32 ビットのアドレス枯渇問題などから IPv6 (Internet Protocol version 6) [2] が利用され始めている。

インターネットの経路制御は、図 2.1に示すように階層化が成されている。

経路制御においてもっとも大きな構成要素は、AS と呼ばれる単一の管理権限、管理ポリシーで運用している独立したネットワークシステムである。AS の具体例として、企業の情報システムや学術ネットワーク、商用の ISP などがある。AS 間の経路制御プロトコルは EGP (Exterior Gateway Protocol) と呼ばれる。EGP の代表的なものは BGP (Border Gateway Protocol)[6] である。それに対し、AS 内の経路制御プロトコルは IGP (Interior Gateway Protocol) と呼ばれる。IGP の代表的なものは、RIP (Routing Information Protocol)[7]、OSPF (Open Shortest Path First)[8] 等である。

一つの AS 内には複数のローカルネットワークが存在する。一般的なユーザは、ISP と契約し、ISP に接続されたローカルネットワークを利用しインターネットへの接続を行う。このように、インターネットは大小さまざまなネットワークが階層的に組み合わされている。

本研究では、この階層的なインターネットの中で、最も末端に位置するネットワークの事を**「終端ネットワーク」**と呼ぶ。具体的には学校や ISP のローカルネットワークのことであり、ユーザが直接接続するネットワークのことである。また、現在利用されているローカルネットワークは、そのほとんどが Ethernet[9] を利用している。そのため、本研究の終端ネットワークは Ethernet によって構築されていることを前提とする。終端ネットワークにおける経路制御の特徴を次節で述べる。

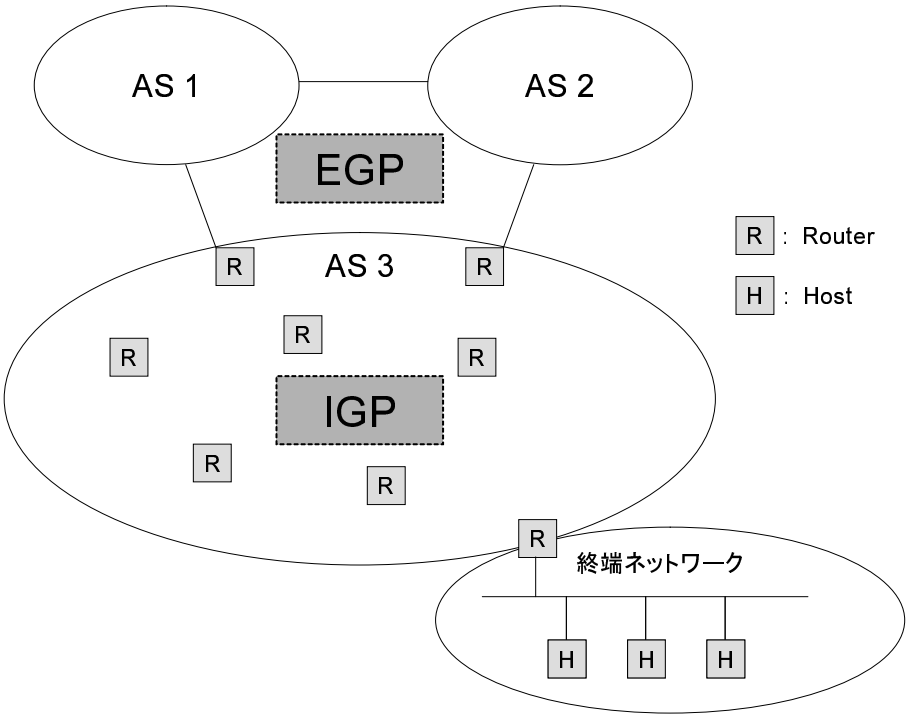

図 2.1: 経路制御の階層化

2.2　終端ネットワークの経路制御における特徴とその問題点

終端ネットワークにはユーザの利用するパソコンなどのノードが利用される。そのため、終端ネットワークにおける経路制御における特徴は、ユーザが利用するノードの特徴に依存する。

パソコン等、ユーザが直接利用するノードの経路制御における特徴は以下のようになる。

- IP パケットの転送を行わない
  終端ネットワークにある全てのノードは、インターネットにおける末端のノードである。そのため、IP パケットの転送を行う必要がない。

- 経路制御プロトコルを利用しない
  ユーザの利用するノードは通常、各ホストが経路制御プロトコルを利用する際の問題点を考え、経路制御プロトコルを利用しない。問題点には、ノードの処理コスト増加、セキュリティの低下などがある。その代わりに、ネットワーク内にある外部への接続時に利用する単一のルータを指定する。これをデフォルトゲートウェイ (Default Gateway) と呼ぶ。

  デフォルトゲートウェイの指定方法は以下の二通りである。

  – 手動設定
    各ユーザが手動でデフォルトゲートウェイの IP アドレスを指定する。しかし、大規模なネットワークでは、全てのユーザに利用するネットワークの状況を把握させること非常に困難である。
  – 自動設定
    ユーザにデフォルトゲートウェイの IP アドレスを意識させることなく、自動的に設定さ

せる。一般的に IPv4 ネットワークでは DHCP (Dynamic Host Configuration Protocol) [10] を利用する。また、IPv6 ネットワークでは NDP (Neighbor Discovery Protocol)[5] を利用する。

大規模なネットワーク等で全てのユーザに、そのネットワークのデフォルトゲートウェイの IP アドレスを知らせ、手動で設定させるのは難しい。多くのネットワークでは自動設定が用いられている。

このように、終端ネットワークにある各ノードは、単一のデフォルトゲートウェイルータを指定し通信を行っている。そのため、デフォルトゲートウェイルータに障害が発生した場合、その終端ネットワーク内のホストが全て外部への接続を断たれてしまうという問題点がある。

このような問題を、**終端ネットワークにおけるシングルポイント障害 (single point of failure)** と呼ぶ。シングルポイント障害の問題を解決するためには、CiscoSystems[11] の HSRP (Hot Standby Router Protocol)[12] や、 VRRP を利用することが現在一般的である。

第3章　VRRP: Virtual Router Redundancy Protocol

本章では、IPv6 用に設計された VRRPv3 (Virtual Router Redundancy Protocol version 3)[13] の詳細を述べる。

3.1　プロトコル概要

3.1.1　目的

通常の終端ネットワークでは、全てのノードは単一のデフォルトゲートウェイを利用し、インターネットへの接続を行っている。そのため、デフォルトゲートウェイに障害が発生した場合、その終端ネットワークに接続している全てのホストが、外部への接続を断たれてしまう。

VRRP は、終端ネットワークにおけるシングル・ポイント障害を解消するためのプロトコルである。

3.1.2　動作概要

VRRP は複数の物理的なルータによって一つの仮想ルータを構成する。一つの仮想ルータを共有するルータのグループを VRRP グループと呼ぶ。その中の一台がマスターとなり、それ以外はバックアップとなる。そして、正常状態ではマスタールータが IP パケットの転送を行う。マスタールータはグループ内の全てのルータに対し生存広告パケットを定期的に送信する。バックアップルータは、生存広告パケットを受信しマスターの状態を監視しておく。マスターに障害が発生した場合は、バックアップの中の一台がマスターの状態を引き継ぎ、IP パケットの転送を継続する。

バックアップが引き継ぐマスターの状態は、リンクローカルアドレス [14][15] と MAC アドレスの組である。リンクローカルアドレスとは、ローカルネットワーク内のみで有効なアドレスである。fe80::で始まり、その後に MAC アドレスから自動的に生成された識別子をつける。別ネットワークにあるノードとの通信にはグローバルアドレスを利用し、同一ネットワーク同士ではリンクローカルアドレスを利用し通信を行う。ホストは、デフォルトゲートウェイのリンクローカルアドレスと MAC アドレスをルータ広告によって取得し、その取得した組に向けて通信を行う。VRRP グループはこの組を共有しているため、ホストはルータの切り替えを意識する必要がなく、通信を継続できる。

VRRP グループに属する VRRP ルータは、それぞれ 1 から 255 の優先度値 (Priority) が定められている。グループ内で最も高い優先度値を持つルータがマスタールータとなる。最大値である 255 を持つルータは、グループで共有する IP アドレスの元々の所有者である。

3.2　状態遷移

VRRP は図 3.1に示す状態遷移に従って動作する。

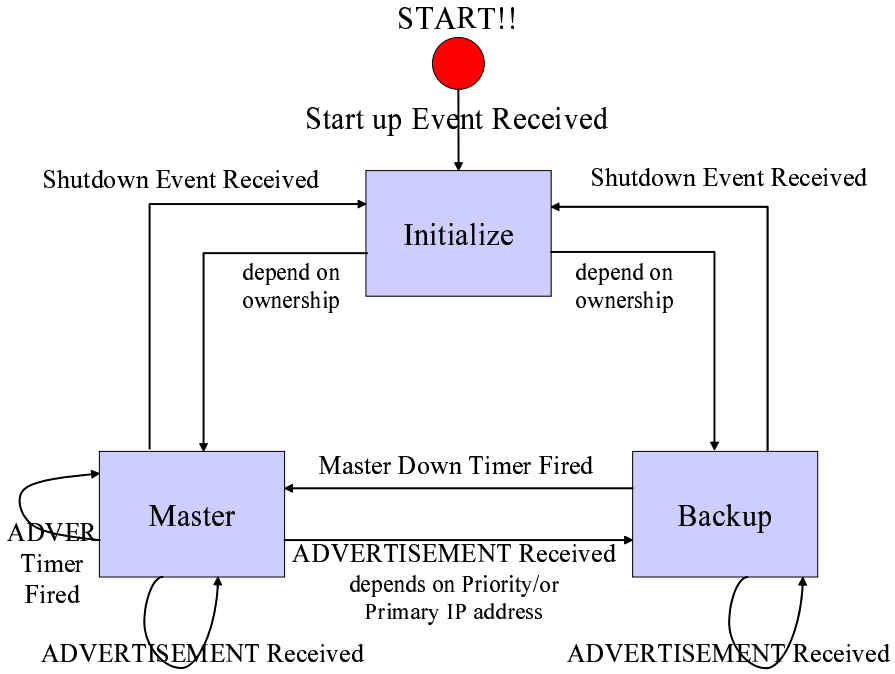

図 3.1: VRRP の状態遷移

本節では、各状態の目的、動作を説明する。

3.2.1　初期化 (Initialize)

初期化状態は、起動時等に自らがマスター状態になるかバックアップ状態になるかの判断を行う状態である。この状態では、自らの優先度値を確認し、それによってマスターかバックアップかの判断を行う。

優先度値の最大値である 255 は、そのルータが元々所有しているリンクローカルアドレスを、VRRP グループで共有している場合である。そのため、255 であれば無条件にマスターになる。初期化状態からは、優先度値 255 のルータのみがマスターへと推移する。

逆に優先度値が 255 以外の場合は、バックアップ状態へと推移し、マスタールータからの生存広告パケットの受信を開始する。

3.2.2　マスター (Master)

- 基本動作

  マスター状態は、グループ内で唯一実際に IP パケットの転送を行うルータである。VRRP グループの共有する IP アドレス、MAC アドレスを利用して、デフォルトゲートウェイルータとしての動作を行うのがマスター状態である。

そのため、次に示す動作を行う必要がある。

- 仮想ルータの持つ IPv6 リンクローカルアドレスの要請ノードマルチキャストアドレスへ参加する
- 仮想ルータの持つ IPv6 アドレスへの ND 近隣要請に返事をする
- ルータ要請に返事をする
- 宛先 MAC アドレスが仮想ルータの MAC アドレスである IP パケットの受信および転送

- 生存広告パケットの送信

  マスター状態になった VRRP ルータは、VRRP 生存広告パケットの送信を始める。通常、送信の間隔は 1 秒である。この広告の目的は、他の VRRP ルータに自らがマスター状態であることを伝え、他のルータをバックアップへと推移させることである。そして、このパケットを送るルータはグループ内で唯一のマスタールータとなる。逆に、VRRP グループ内の別のルータからの生存広告パケットを受信した場合は、そのパケットの優先度値を確認する。もしその値が自らの優先度値よりも高い場合は、バックアップ状態へと移行する。

- 非要請 ND 近隣広告の送信

  ルータはマスターに推移した際、NDP の非要請 ND 近隣広告の送信を行う。これは、途中ネットワーク上にあるスイッチングハブの MAC アドレスキャッシュを更新するためである。

  スイッチングハブはリピータハブと異なり、各ポートにどの MAC アドレスのマシンが接続されているかのキャッシュを持つ。そして、キャッシュに保存されている MAC アドレス宛のパケットを転送する際、そのキャッシュに指定されたポートにのみ転送する。

  VRRP では VRRP グループ内で単一の MAC アドレスを共有し、マスターのルータが利用する。マスターであるルータが、他のポートに接続されている別のルータに切り替わった際、スイッチングハブ上の MAC アドレスのキャッシュを上書する必要がある。それを行わないと、パケットは正しいポートに転送されず、マスタールータへ送られない。そのため、正しい転送を行うために ND 近隣広告を送信し、キャッシュを上書きする。

  非要請 ND 近隣広告を送信するマシンはルータであり、近隣要請に対する応答ではなく、ホストの近隣キャッシュを上書きするものである。そのため、このパケットの Router flag と Override flag をセットし、 Solicited flag はセットしない。また、パケット内の Target Link Layer アドレスには VRRP グループが共有している MAC アドレスの値を挿入する。

- パケットの受信

  IETF (Internet Engineering Task Force)[16] の VRRP ワーキンググループでは、VRRP グループが共有する IPv6 アドレス宛の IP パケットの受信処理について、二通りの方法が議論されている。

  一つ目は、自らがそのアドレスの所有者か否かで動作を変更する方法である。その目的はネットワーク管理者等が VRRP ルータと通信を行う際に、障害発生の検知を容易にするためである。アドレス所有者である場合 (優先度値が 255) はそのパケットを受信、処理をする。所有者でない場合 (優先度値が 255 以外) はパケットを破棄し、処理しない。通常、ネットワークの障害判断は PING というツールで行う。PING は、ICMP パケットを送信し、その返答から相手ホストの生存を確認するツールである。アドレスの元々の所有者ではないルータが

通信を行ってしまうと、バックアップルータが PING に対して返答を行う。そのため、マスターとバックアップの判断を行うことが不可能になり、バックアップに切り替わった事を検知することが難しくなる。

二つ目は、アドレスの所有者ではないルータも仮想 IP アドレス宛のデータを受信する方法である。ホストを接続するユーザは、デフォルトゲートウェイルータに対して、PING などのツールを用い生存の確認を行う場合がある。この時に、返信を行わないとルータが動作していないと判断し、通信を諦めてしまう可能性がある。そのため、仮想 IP アドレス宛のパケットに対してのデータも受信し、通信する。

従来は一つ目が採用されていたが、障害の検知は他の方法でやるべきであるなどの理由により、現在は二つ目が採用されている。

3.2.3　バックアップ (Backup)

バックアップ状態は、VRRP グループ内にあるマスタールータを監視することが目的である。

バックアップ状態は、マスターからの生存広告パケットを待ち続ける。一定時間待ち続けパケットが来なかった場合、マスターに障害が発生したと判断し自らがマスター状態になる。一定時間内に広告パケットが到達した場合はマスターが生存していると判断し、改めて一定時間待ち続ける。ただし、受信したパケット内の優先度値が自らの優先度値より低い場合は、そのパケットを破棄する。

バックアップ状態は、VRRP グループ内ではルータとして動作しない。

3.3　生存広告

マスター状態の VRRP ルータは、同一 VRRP グループに所属する VRRP ルータに対して自らの生存を広告するマルチキャストパケットを定期的に送信する。広告を受け取るマルチキャストアドレスには、同一 VRRP グループに所属する全ての VRRP ルータが参加している。

マスタールータは、生存広告パケットを、広告インターバルに従い定期的に送信する。バックアップルータは、一定時間広告パケットが届かないことを確認すると、マスタールータに障害が発生したと判断する。そしてマスタールータの状態を引き継ぎ、自らがゲートウェイルータとして転送を始める。各 VRRP ルータには 1〜255 までの優先度 (Priority) の値が設定されている。バックアップルータには各々が持つ優先度値を利用しマスター障害判断の計算を行う。このとき、優先度が高い VRRP ルータほど、障害判断までの時間が短くなっている。そのため、原則的に優先度の高いルータが低いルータよりも早く障害判断を行い、マスターになる。

障害発生判断の計算方法は具体的には以下のようになる。

(3 × 広告インターバル) + ((256 - 優先度) / 256)

VRRP の広告パケットは図 3.2に示すカプセル化がなされている。
各フィールドは次のようになっている。

- IPv6 Header

図 3.2: 生存広告パケットのカプセル化

– 発信元アドレス
パケットを送信するインターフェイスのリンクローカルアドレス。

– あて先アドレス
IANA(Internet Assigned Numbers Authority) から VRRPv3 用に割り当てられた次のアドレスの中の一つを利用する。

ff02::1 - ff02::ffff:ffff

ただし、draft-ietf-vrrp-ipv6-spec-05 には以下のように書かれている。

```
A convenient assignment of this link-local scope multicast would be:

      FF02:0:0:0:0:0:0:12

  as this would be consistent with the IPv4 assignment for VRRP.
```

– ホップリミット
255 に設定されていないとならない。広告パケットはそのローカルネットワーク内のみで交換されないといけない。外部からの、ルータを経由したパケットはホップリミット値が 255 以下になるのでこの条件によって外部からのパケットを無視することが可能になる。

– Next header
IANA から割り当てられた 112 という番号を利用する。

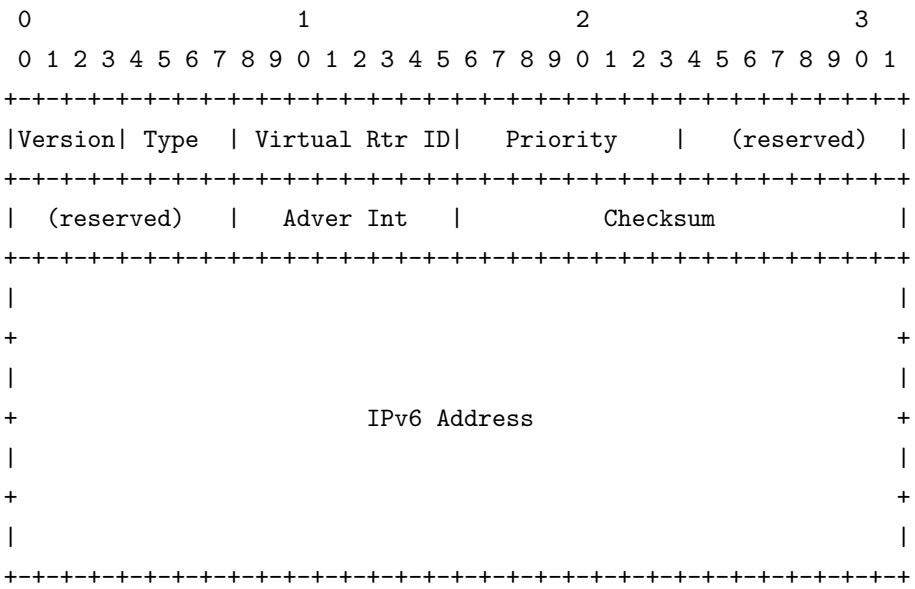

図 3.3: VRRP パケットフォーマット

- VRRP パケット

  VRRP パケットフォーマットを図 3.3に示す。

  - バージョン
    VRRP for IPv6 のバージョンは３である。受信した時に、正しくない場合、破棄する。
  - タイプ
    ”１-生存広告”というタイプのみが定義されている。受信した時に、１でない場合、破棄する。
  - VRID
    バーチャルルータが所属している VRID。同受信した時に、同じ VRID でない場合、破棄する。
  - プライオリティ
    パケットを送信するルータのプライオリティ。
  - 広告インターバル
    マスターに設定された広告インターバル。受信した時に、バックアップに設定されていたものと同じでない場合、破棄する。
  - チェックサム
    IP ヘッダから VRRP パケットまでのチェックサム。受信した時に、この値が正しくない場合、破棄する。
  - IPv6 アドレス
    VRRP グループが共有している IPv6 アドレス。

3.4　仮想インターフェイス

VRRP では、エンドノードに対してルータの切り替えを意識させないために仮想インターフェイスという概念を用いている。仮想インターフェイスは、仮想 IP アドレスと仮想 MAC アドレスの組を VRRP グループ内で共有することによって実現されている。各 VRRP グループは単一の仮想 IP アドレスと仮想 MAC アドレスを持ち、グループ内でマスターのルータのみがこのアドレスを利用し、転送する。IP アドレスと MAC アドレスが組で別ルータに移動するため、エンドノードからはその移動が意識されることがない。よって円滑な切り替えが可能になる。

VRRPv3 では、仮想 IP アドレス、仮想 MAC アドレスは以下のように定められている。

- 仮想 IP アドレス
  そのローカルネットワーク内のリンクローカルアドレスを利用する。仮想 IP アドレスの本来の持ち主である VRRP ルータは、優先度を 255 にしなければならない。
- 仮想 MAC アドレス
  次のアドレスが VRRPv3 用に割り当てられた MAC アドレスである。

  00-00-5E-00-02-(VRID)

  最初の 3 オクテット (00-00-5E) は IANA が VRRP に割り当てたものである。次の 2 オクテット (00-02) は VRRPv3 に割り当てられたものである。最後の 1 オクテットを VRID にすることで、1〜255 まで 255 種類の MAC アドレスが用意されている。

3.5　Preempt モード

VRRP の追加機能である Preempt モードに関する説明をする。

マスター状態のルータよりもプライオリティの高いバックアップルータがある場合、どのように動作するかの処理をこの値によって変更する。Preempt モードが真の場合、常に優先度が高いルータをマスターにする。偽の場合は優先度の高いルータが後から出現した場合でもマスタールータの変更はない。また、特に設定の無い場合、Preempt モードは真となる。

ただし、優先度が 255 の場合は Preempt モードが偽であっても常にマスター状態になる。優先度が 255 とは、「その VRRP グループが共有する IP アドレスの本来の持ち主である」という意味である。そのため、持ち主であるルータが正常な場合は Preempt モードに関係なく最優先でマスターになる。

Preempt モード真偽の違いを具体的に実現させるためには一箇所だけ動作を変更する。バックアップ状態で広告パケットを受け取った時に、Preempt モードが真であれば、受信したパケットの優先度が自らの優先度より高い場合のみパケットを処理する。Preempt モードが偽の場合は受信したパケットの優先度に関係なく、パケットを処理する。

3.6　負荷分散

VRRP の複数ルータ全てに障害が無い場合、その中から単一のルータのみを利用するのではなく、同時に多数のルータを利用する方法がある。この方法を用いる事によってルータの処理量や回

線へのや負荷が分散され、処理性能が向上する。IETF の VRRP ワーキンググループが提出した、技術提案書である Internet-Draft には以下のような方法が示されている。

図 3.4のように一つのネットワーク内に複数のグループを作成する。複数の VRRP グループが存在するとき、お互いは全く独立に動作する。図の例では、VRID 1 のグループのマスタールータはルータ R1、バックアップルータはルータ R2 と設定してある。逆に、VRID 2 のグループのマスタールータはルータ R2、バックアップルータはルータ R1 とする。そして、複数あるエンドノードを複数有る VRRP グループに分けて所属させる。具体的には設定されるデフォルトゲートウェイの IP アドレスが異なるようにする。図の例では、ホスト 1、2 はデフォルトゲートウェイを R1 の IP アドレスにしているため、VRID1 に所属していることになる。逆にホスト 3，4 は VRID2 に所属している。

上記のようにネットワークを構成することにより、ルータに障害が無い場合、所属する VRID のマスタールータを利用し通信が行われる。そして、結果としてトラフィックを分散させることが可能になる。また、どちらかのルータに障害が発生した場合でも本来の VRRP の機能として通信の継続をすることが可能である。

しかし、ここに示されている方法は使われていない。複数あるホストそれぞれに別々のデフォルトゲートウェイを設定するのがこの方法なのだが、この設定を行う事が困難なためである。デフォルトゲートウェイは通常、IPv4 の場合は DHCP、IPv6 の場合は NDP を利用して自動的に設定される。現在ある実装では、DHCP は単一のデフォルトゲートウェイアドレスのみを返す。また、NDP も各ルータが広告出来るデフォルトゲートウェイアドレスは一つである。複数のルータから別々のアドレスの NDP を投げた場合の動作はクライアントの実装に依存する。このように、複数あるクライアントに別々のデフォルトゲートウェイアドレスを自動で設定することは現在不可能である。もしこの方法を用いる場合は、DHCP サーバに、VRRP との連携によってリクエストに対して返す値を振り分ける等の変更を加える必要がある。

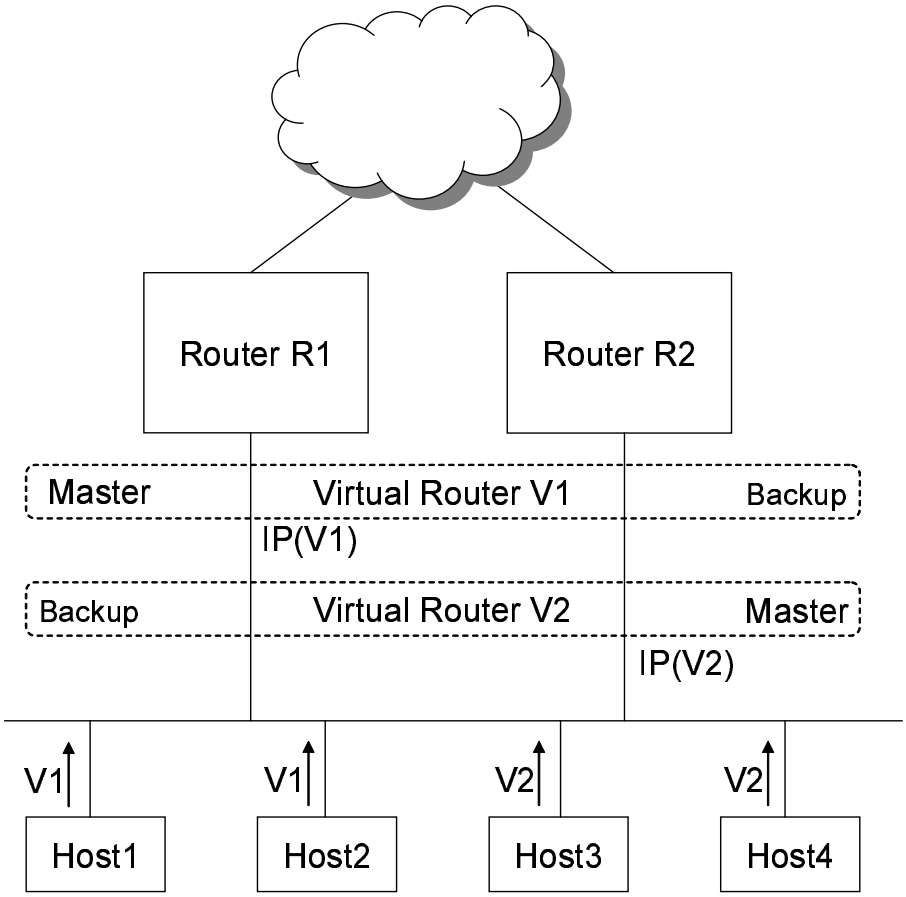

図 3.4: VRRP における負荷分散

第4章　問題点と解決へのアプローチ

本章では、終端ネットワークにおける現状の問題点を整理し、その解決アプローチを述べる。終端ネットワークにおける障害を定義し、その対応および解決できない障害について議論する。本研究では、下流ネットワークで障害が発生した場合に着目し、その解決アプローチを述べる。

4.1　VRRP と経路制御プロトコルの同時利用

VRRP は、ホストのデフォルトゲートウェイルータに障害が起こった際、それを有効なルータに切り替えるプロトコルである。そのため、終端ネットワークから外向きのトラフィックには有効であるが、内向きのトラフィックの事は考えられていない。一般的なネットワークでは、デフォルトゲートウェイルータは、上流のルータ群と経路制御プロトコルを利用し、経路の選択を行っている。内向きの経路の選択は経路制御プロトコルによって行われている。つまり、終端ネットワークに対する内向きの経路は、経路制御プロトコルによって冗長化が成されている。

4.2　終端ネットワークにおける障害の定義

終端ネットワークにおける障害箇所を図 4.1を用いて分類する。図中の”host”はホストを表しており、ホストが接続されているネットワークが終端ネットワークである。図中の OSPF は、ルータの上側と下側それぞれのインターフェイスを監視している。また、VRRP・OSPF 共にルータ A を利用するような設定が行われている。

図中の◯で表現されている具体的な障害箇所は次のように定義される。

① vrrp ルータの上流 OSPF ネットワーク内の回線に障害が発生

② vrrp ルータの上流のインターフェイスに障害が発生

③ vrrp ルータ本体に障害が発生

④ vrrp ルータの下流のインターフェイスに障害が発生

⑤ vrrp ルータの下流ネットワーク内の回線に障害が発生

⑥ vrrp ルータの下流ネットワーク内の回線に障害が発生 (ホストが分断)

4.3　現状の対応とその問題点

本節では、各障害箇所における現状の対応と、現状で問題となっている点について述べる。

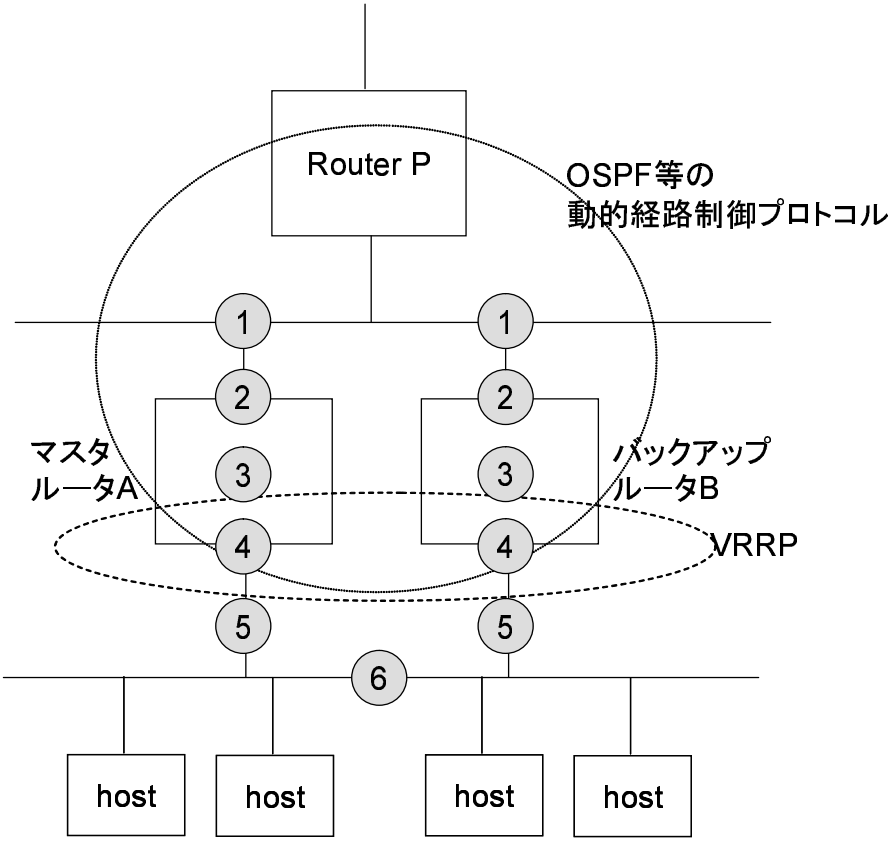

図 4.1: 障害発生箇所

4.3.1　各障害箇所における対応

- 障害箇所 ① における対応

  VRRP はルータの内側のネットワークで広告を送信し、障害判断を行っている。そのため、VRRP でこの箇所に発生した障害を検知することは不可能である。

  OSPF はルータ A が上流インターフェイスを利用し外部に接続できない事を判断する。そして、下流のインターフェイスを通りルータ B を経由した代替経路へと経路変更する。また、内向きの経路はルータ A からルータ B へと切り替えられれる。そのため、下流のホストが外部への接続を行う場合、ルータ A を通った後にルータ B を経由する一ホップ遠回りな経路を通ることになる。

  また、ルータの ICMP Redirect[17] が有効な場合は、ホストの向ける経路ががルータ B へと書き換えられる。そのため、ホストは VRRP の仮想 IP アドレスを利用しなくなるため、障害から復旧した後もルータ A に経路を切り替えないという問題が発生する。

- 障害箇所 ② における対応

  ①の場合と同様、OSPF によって経路がルータ B を経由するものへと切り替えられる。

- 障害箇所 ③ における対応

ルータ本体に障害が発生した時は、OSPF、VRRP 共に障害検知が可能である。そのため、冗長経路へ収束することが可能である。

- 障害箇所 ④ における対応

  OSPF は下流側ネットワークの障害検知をすることが不可能である。そのため、正しい経路を発見することが出来ない。

  VRRP ではインターフェイスに障害が発生した場合、そのルータは生存広告を送信することが不可能になる。よって VRRP グループ内の別の正常なルータがマスターになる。

  そのため、内向き経路にブラックホールが発生してしまい通信を継続出来ない。

- 障害箇所 ⑤ における対応

  OSPF は下流側ネットワークの障害検知をすることが不可能である。そのため、正しい経路を発見することが出来ない。

  VRRP は、マスタールータ A からの広告がバックアップルータ B に届かなくなる。そのため、ルータ B がマスターに状態遷移する。また、ルータ A も広告を送信し続けるので、両ルータがマスターという状態になる。両ルータ共にマスターということは両ルータ共に転送を行うということである。そのため、どちらの障害発生時も外向きのトラフィックが途絶えることはない。

  そのため、内向き経路にブラックホールが発生してしまい通信を継続出来ない。

- 障害箇所 ⑥ における対応

  ⑤の場合と同様、両ルータがマスターとなり、外向きの経路は確保できる。しかし、OSPF で分断されたホスト全てに経路を向けることは不可能である。

4.3.2　現状の問題点

上記した問題点をまとめると、以下の 3 種類の問題が発生している。

1. 上流障害時の問題

   上流で障害が発生した際、OSPF によって内向き経路外向き経路共に回復できる。しかし、外向きの経路が二台のルータを通る無駄なものになってしまう。

2. 下流障害時の問題

   下流で障害が発生した際、VRRP によって外向き経路は回復できる。しかし、内向き経路には変更が起こらず、通信が継続出来ない。

3. セグメント分割

   終端ネットワーク内の途中回線に障害が発生した場合、ホストが分断されてしまう。OSPF は一つのプレフィックス向けの経路を複数のルータに向けることが出来ないため、全てのホストの通信を継続することは不可能である。

4.4　解決へのアプローチ

本研究では、上記の問題のうち、下流障害時の問題に着目し、OSPF と VRRP の連携というアプローチによって解決する。

具体的には、VRRP によって判断された障害情報を OSPF へ伝えるために、経路情報の再配布を利用する。VRRP ルータが OSPF へ下流のプレフィックスへの経路情報の再配布する。マスター状態の時にのみ経路情報を持ち、逆にそれ以外の場合の時はその経路情報を削除する。この方法によって OSPF の終端ネットワーク向けの経路を VRRP でマスター状態のルータへと向けることが可能である。

第5章　VRRPデーモンの設計

本章では、VRRP デーモンの設計について述べる。

5.1　設計要件

本研究で作成する VRRP デーモンは、経路制御プロトコルとの連携処理が必要である。しかし、経路制御プロトコルは多くの種類があり、それぞれに対して個別の連携処理を行うことは、処理効率が悪い。また、新たな経路制御プロトコルへの対応も不可能である。

本研究では、統一的にカーネルや経路制御プロトコルへの接続を行うシステムを利用する。そして、そのシステムに対して統一的なインターフェイスで各プロトコルやカーネルへの接続を行う。

5.2　Zebra

本研究では、VRRP の実装を Zebra[18] パッケージ上で行う。

GNU Zebra は経路制御ソフトウエアパッケージで、FreeBSD、NetBSD、Linux 等の代表的なフリー UNIX OS 上で動作する。サポートしているルーティングプロトコルは RIP、RIPng、OSPF、OSPFv3、BGP 等である。

また、他のルーティングソフトウェアに比べ次のような特徴を持っている。

- フリーソフトウェア
  Zebra は GPL (General Public Licence)[19] ソフトウエアで、オープンソースである。この利点は、世界中のユーザが自由に利用出来ることや、世界中の開発者が開発に携われることである。そのため、問題発見などが非常に早く、安定したソフトウェアを作ることが可能である。

- 動作の安定性
  Zebra は世界的に普及しているソフトウエアであり、安定性が保障されている。

- ユーザインタフェイス
  Zebra には、ネットワーク機器メーカであるシスコ・システムズ (Cisco Systems) 社の IOS (Internetwork Operating System) というルータ用の OS に似せた対話型のユーザーインターフェイスが提供されている。そのため、多くのユーザが違和感なく Zebra を利用することが可能である。

- プロトコル毎に独立したプロセス
  図 5.1で示すように、各プロトコルは Zebra 本体とは独立した、１プロトコル１プロセスで動作する。各プロトコルデーモンはそのプロトコル処理のみを行ない、プロセス間通信によって渡された情報を元に zebra デーモンがカーネルに対して設定の変更を行なう。この利点は、

新しいプロトコルを追加する際にソースコードの変更が少なくてすむ点や、保守を容易にする点などが挙げられる。

本研究で VRRP を実装するにあたり、上記の利点以外にも次の点を重視し Zebra 上での実装を選択した。

- VRRP とルーティングプロトコルの連携

  4章で示したように、VRRP は上流のルーティングプロトコルと連携するべきである。そのためには、多くのルーティングプロトコルを実装している Zebra は、ルーティングプロトコルとの連携を実装する際に非常に有効である。また、各プロトコル毎に独立したプロセスである点からも、プロトコル間の連携を単純にすることができる。

- VRRP とルータ広告の連携

  Zebra はルータ広告の送信を行うことが可能である。

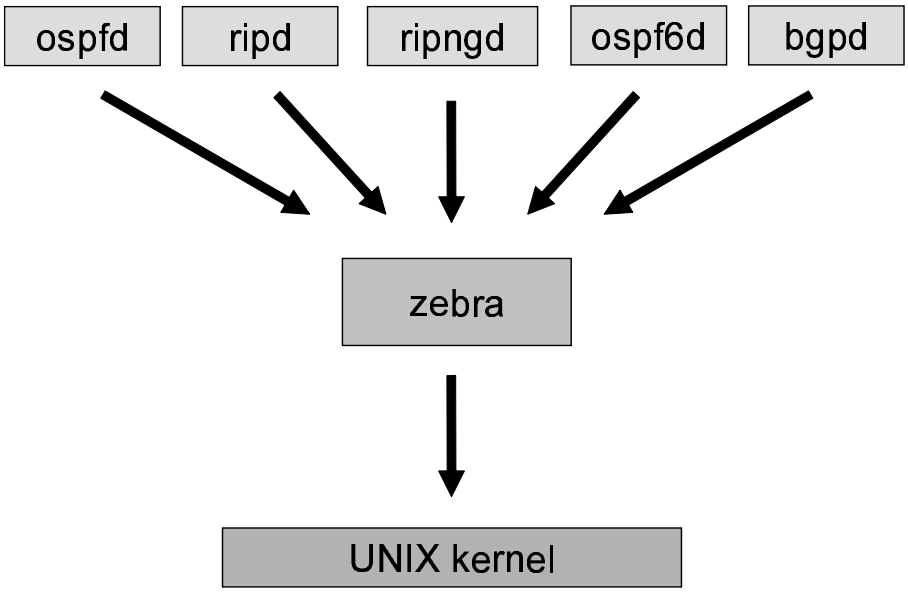

図 5.1: Zebra プロセス図

5.3　ルーティングプロトコルと VRRP の協調

5.3.1　経路再配布による協調

下流側の VRRP がバックアップになっているルータは、終端ネットワーク内への経路が途切れている可能性がある。そのため、内向きの経路も、VRRP でマスターになっているルータを利用しなくてはならない。

本システムでは、VRRP デーモンで持つ下流の経路情報を Zebra を介し経路制御プロトコルへ再配布する。VRRP はマスター状態の時にのみ経路を保持し、バックアップ状態では経路を消す。

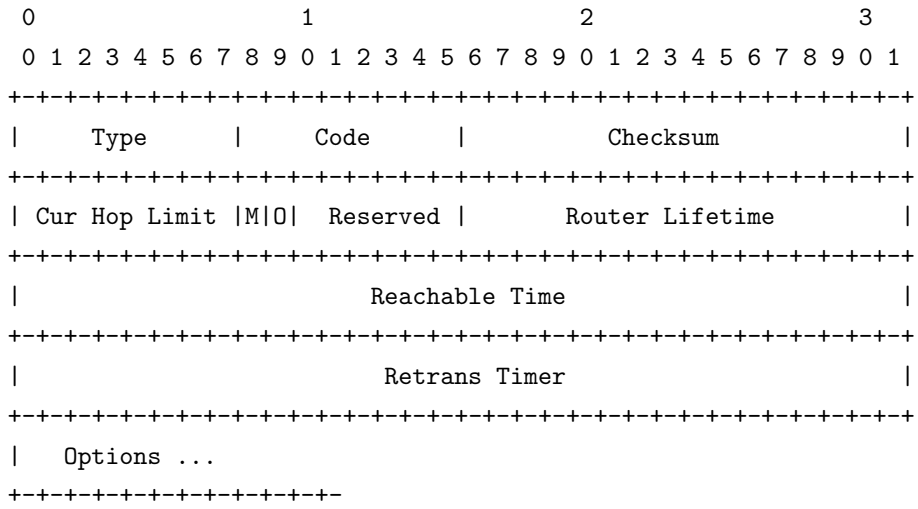

図 5.2: ルータ広告パケットフォーマット

その結果、マスタールータのみ経路の再配布を行い、経路制御プロトコルからの内向きの経路を常にマスタールータに向けることが可能になる。

5.3.2　ICMP Redirect の無効化

マスタールータの上流回線に障害が発生すると、ルーティングプロトコルによってマスタールータの外向き経路が、バックアップルータの下流側のインターフェイスに向けられる。そのため、ホストから外部への経路が、「ホスト→マスタールータ→バックアップルータ→外部ルータ」と、同一セグメント内のルータを二度通ることになる。そして、マスタールータから ICMP Redirect が送信され、ホストの経路がバックアップルータの下流インターフェイスへと向けられる。

ホストのデフォルトゲートウェイ経路は VRRP の仮想 IP アドレスに向けられている。しかし、ICMP Redirect によって、経路がバックアップルータの物理 IP アドレスに書き換えられてしまう。そのため、それ以降、VRRP を用いたルータの冗長化が不可能になってしまう。

本システムでは、この問題を防ぐために、VRRP を有効にした際は ICMP Redirect を無効にする。

5.4　ルータ広告と VRRP の協調

ルータ広告 (RA : Router Adversisement) は IPv6 ネットワークにおいて、ルータがホストに対して、自らがデフォルトルータであることを広告するパケットである。VRRP は VRRP グループの中で、転送を行うマスタルータがルータ広告の送信を行う。そのため、VRRP の状態によって、ルータ広告の送信を制御する必要がある。

5.4.1　ルータ広告動作概要

ルータ広告のパケットフォーマットを図 5.2に示す。

ルータはネットワーク上の全てのホスト・ルータが所属する、全ノードマルチキャストアドレス (ff02::1) へルータ広告パケットを送信する。
ホストは、自らのグローバルアドレス、デフォルトゲートウェイ等の設定を行う。
ルータ広告の送信パターンは以下の二通りである。

- 定期的な送信
  ルータは定期的にルータ広告を送信する。
- 受動的な送信
  ホストは、はじめにネットワークに接続した時に、ルータ要請 (RS : Router Solicitation) パケットを送信する。このパケットを受信したルータはそれを受け、ルータ広告を送信する。

5.4.2　協調方法

VRRP がルータ広告の送信を制御する時には以下のような動作をする。
上に示したパケットフォーマットやそれ以外の情報から、VRRP システムがルータ広告システムに与える情報は以下のようになる。

- ルータ広告を送信するか、否か
  ルータ広告はそのルータがアクティブである時に、ホストにそれを伝えるためのものである。そのため、VRRP ルータがマスターの時はルータ広告を送信し、それ以外ではルータ広告の送信を止める必要がある。
- 通知する IPv6 アドレス・リンク層アドレス
  VRRP ルータは単一の仮想 IP アドレス・仮想 MAC アドレスを複数ルータで共有する。そのためホスト群からは単一のルータとして見せることが可能になっている。よって VRRP システムからルータ広告システムへ、通知する仮想 IP アドレス・仮想 MAC アドレスを与えなくてはならない。
- インターフェイス名
  ルータ広告パケットをどのインターフェイスから送るかを指定しなくてはならない。

5.5　仮想インターフェイスの実現

5.5.1　疑似インターフェイス概要

疑似インターフェイスは図 5.3に示すように物理インターフェイスとの関係を持つ。各疑似インターフェイスには対応する物理インターフェイスを設定する。また、各々が IP アドレスや MAC アドレスを持つことが可能である。そして、疑似インターフェイスはその IP アドレスや MAC アドレスを利用して通信を行うことが可能である。
実際に通信を行うのは物理インターフェイスである。疑似インターフェイスに対応付けられた物理インターフェイスで受信したパケットは、カーネルによって宛先 MAC アドレスを確認される。その宛先 MAC アドレスが疑似インターフェイスの MAC アドレスと一致した場合は疑似インターフェイスが受信したものとして処理される。疑似インターフェイスから出力する場合は、疑似インターフェイスの MAC アドレスを発信元 MAC アドレスにし、物理インターフェイスから出力する。

図の例では、物理インターフェイス P を利用する VRID は二つある。それぞれの設定を行う疑似インターフェイスが疑似インターフェイス A と B である。

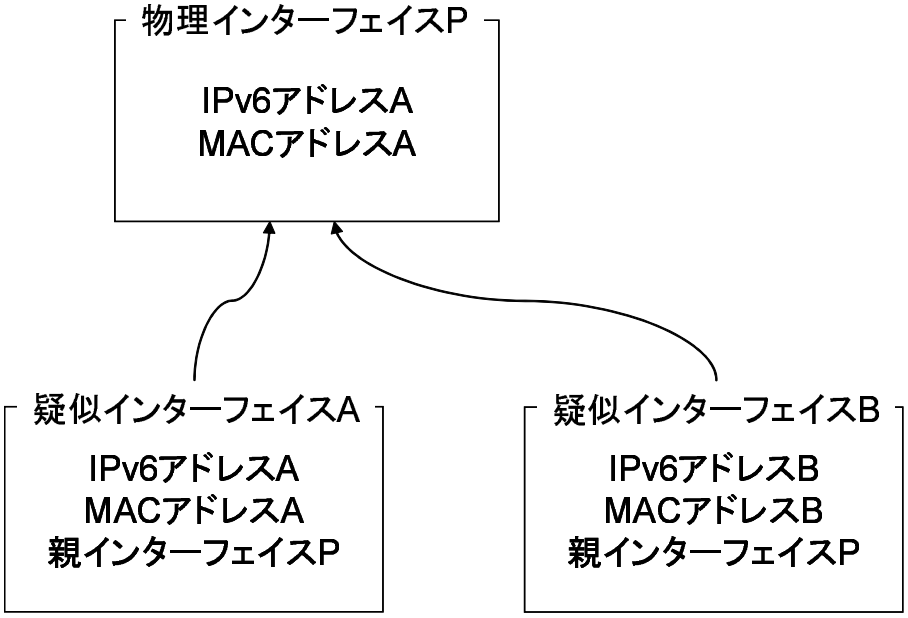

図 5.3: 疑似インターフェイスと物理インターフェイスの関係

5.5.2　利点

この方法を用いる利点は次のようになる。

- 独立した設定情報の管理

  本システムは、VRRP に関する設定は全て疑似インターフェイスに対して行う。マスタ、バックアップ、初期化の三種類の各状態を、疑似インターフェイスの状態を変更することによって実現する。疑似インターフェイスを用いることで、物理インターフェイスには変更は必要ない。そのため、物理インターフェイスに対する既存の設定をそのまま残すことが出来る利点が生まれる。

  マスタ状態の時は、疑似インターフェイスを有効にし、それを介し転送を行う。そして、疑似インターフェイスを無効にすることで、ルータは、バックアップ状態、あるいは初期化状態へと遷移可能である。

- 複数 VRID の実現

  VRRP は、複数の VRID を利用してたすきがけのように運用を行うことがある。(第 3.6節参照) そのため、一つのインターフェイスが複数の仮想 IP アドレスと MAC アドレスを管理しなくてはならない。しかし、物理インターフェイスは単一の MAC アドレスのみを管理する。そのため、疑似インターフェイスを利用せずに複数 VRID を実現することが難しい。promiscuous モード等の方法を用いることによって実現可能だが、非常に処理コストが高い。

単一の物理インターフェイスを利用する疑似インターフェイスを複数作ることによって、複数 MAC アドレスの利用が可能になる。また IP アドレスに関しても、物理インターフェイスは自らのアドレスの管理のみを行えばよい。

- 通信と転送の区別

  VRRP では通信と転送の区別がなされている。VRRP によって共有されているアドレスは転送のみに利用されるアドレスであり、通信には用いられない。共有されているアドレスを用いて通信が可能なのは自らがそのアドレスを所有しているルータ (優先度値が 255) のみである。

  疑似インターフェイスを用いると、この区別を容易にすることが可能になる。疑似インターフェイスを介して受信したパケットは転送のみしか行わず、通信を行わないようにする。そして通信は物理インターフェイスのみで行う。

5.5.3　各状態での動作

- マスター

  マスター状態は実際の転送を行う状態である。そのため状態を以下のように設定する。

  – インターフェイスを有効にする。
  – 通信を行う物理インターフェイスの設定を行う
  – アドレスの設定を VRID 共有アドレスに設定 (IP アドレス、MAC アドレス共に)

- バックアップ・初期化

  初期化状態とバックアップ状態は転送を行わない。そのため以下のように状態を設定する。

  – インターフェイスを無効にする。
  – 通信を行う物理インターフェイスの設定を消去
  – アドレスの設定の消去 (IP アドレス、MAC アドレス共に)

本システムの終了時は、疑似インターフェイスに設定された情報を消去し、無効にする必要がある。システム終了後も設定が残ってしまうと誤動作を起こしてしまう可能性があるためである。VRRP インターフェイスは、本システムを動作させるためのインターフェイスであるため、システム動作以外の環境に設定を残す必要がない。

5.5.4　インターフェイス情報の保持

本節では、各ルータのデータ保持方法に関し、データ構造、利用方法、利点を述べる。

- データ構造

  VRRP デーモンでは、インターフェイスの情報を図 5.4、図 5.5に示すデータ構造で保存する。

  全てのインターフェイスの情報は連結リストによって保存されている。その中で、VRRP 疑似インターフェイスは、VRRP ルータが必要なデータを保存する構造体への参照を持っている。保持するデータは VRID、仮想 IP アドレス、仮想 MAC アドレスなどがある。

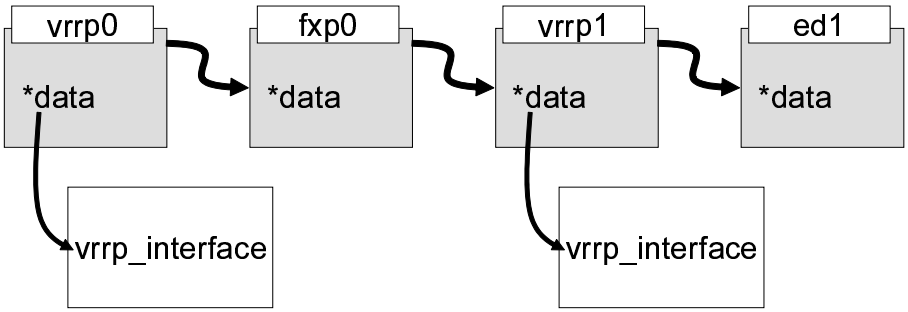

図 5.4: インターフェイスリスト

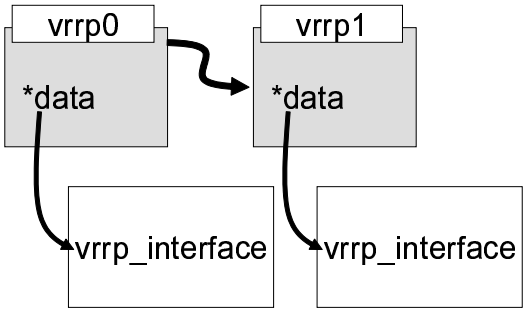

図 5.5: VRRP インターフェイスリスト

また、VRRP 疑似インターフェイスのみの連結リストも用意する。この連結リストも同様に、VRRP データ構造体への参照を持っている。

- 利用方法と利点

  図 5.4のリストは、Zebra デーモンから受信し、保持する。そのため、全てのインターフェイスが含まれている。ユーザが設定を行う際は、このリストから VRRP データ構造体を検索する。そしてユーザが設定の変更を終え、動作を開始する時に図 5.5のリストへデータを追加する。そして、パケットを受信した際は、図 5.5のリストからデータの検索を行う。

  ユーザの設定は、全てのインターフェイスの中から必要なインターフェイスを検索する必要はあるが、その検索頻度は低い。そのため図 5.4のリストの利用が適している。逆に、パケットの受信は VRRP 疑似インターフェイスのみからの検索であり、検索頻度が高い。そのため検索にかかるノード数を減少した図 5.5のリストを用意した。

  このようにデータの保持を二種類に分けることによって、利用用途毎に適した検索を行うことが可能になる。

5.6　パケットの送受信

本節では、本システムが送受信するパケットについてそれらの送信方法を述べる。

5.6.1　VRRP パケット

VRRP パケットは、VRRP デーモンが Raw ソケットを開き、それを利用し送受信する。

疑似インターフェイスは、状態がマスター以外の場合無効にするため、パケットの受信をすることが不可能になってしまう。また、発信元 IP アドレスは物理インターフェイスのリンクローカルアドレスを利用する。以上の理由から VRRP パケットの送受信は、実際に通信を行う物理インターフェイスによって行う。

5.6.2　ルータ広告

ルータ広告は、VRRP デーモンからの指定に従い、Zebra デーモンが送受信を行う。

ルータ広告にはデフォルトゲートウェイのリンクローカル IPv6 アドレスや MAC アドレスの情報を含ませる。また、マスター状態の時のみに送信を行う。以上の理由から、ルータ広告の送信は疑似インターフェイスによって行う。そのことによって、ホスト群に対して、疑似インターフェイスをデフォルトゲートウェイとして広告することが可能になる。

第6章　VRRPデーモンの実装

本章は、実装したVRRPデーモンに関して説明する。

6.1　実装環境

表6.1に示す環境で実装を行った。

使用言語	C言語
OS	FreeBSD4.8Release
KAME patch	kame-20030721-freebsd48-snap
Zebra version	Zebra-0-95pre

表 6.1: 本システムの実装環境

VRRPv3の実装をC言語 [20] にて行った。以下この実装をvrrpdと呼ぶ。vrrpdは、経路制御ソフトウェアパッケージZebra上のに実装した。vrrpdでは、Zebraの機能を利用し、設定の動的な変更、確認が行なえる。OSはFreeBSD 4.8-RELEASE[21] を利用し、その上ではKAMEプロジェクトにより実装されたIPv6スタックが動作している。

6.2　vrrpd の構造

vrrpd のモジュール相関図を、図 6.1示す。また、その動作概要は第 6.3節説明する。

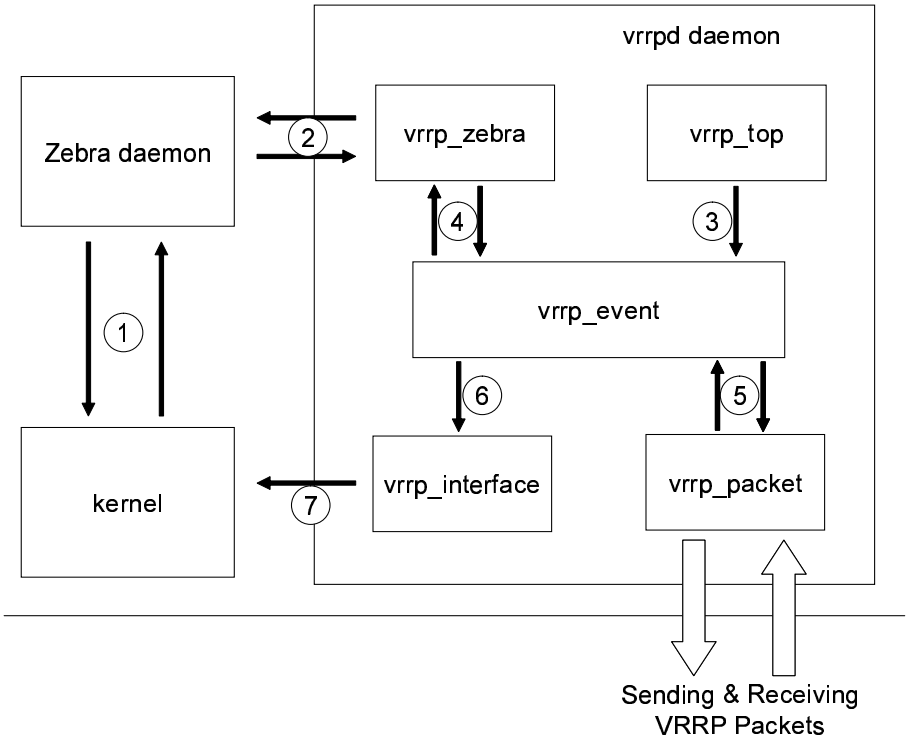

図 6.1: vrrpd のモジュール相関図

```
struct vrrp_interface
{
  struct interface *interface;    /* IF info from zebra */

  struct interface *phy_interface;/* Pointer of physical interface */
  struct in6_addr   vrrp_lladdr;  /* linklocal address to share */
  char              state;        /* -1:Wait, 0:Init, 1:Master, 2:Backup */
  unsigned char     id;           /* Vrid of this interface */
  unsigned char     priority;     /* Priority of this Interface */
  unsigned char     adv_interval; /* Advertisement Interval */

  int               preempt;      /* 1:TRUE, 0:FALSE */
  int               from_master;  /* 1:Accept Master's Address, 0:Not */

  struct ether_addr ether_addr;   /* Virtual Ethernet Address */

  struct thread    *thread_advertise;   /* thread of advertisement */
  struct thread    *thread_master_down; /* thread of master down timer */

  /* Master States */
  unsigned char     m_priority;     /* Master Router's Priority */
  unsigned char     m_adv_interval; /* Master Router's AdvInterval */

  struct in6_addr   m_phy_lladdr;   /* Physical linklocal address */
  struct in6_addr   m_vrrp_lladdr;  /* linklocal address to share */

  /* Additional function */
  struct interface *up_interface;   /* Pointer of upstream interface */
};
```

図 6.2: vrrp_interface 構造体

6.2.1　vrrp_interface モジュール

本システムでは、仮想インターフェイスを KAME パッケージ上に実装された疑似デバイス (疑似インターフェイス) の利用によって実現する。vrrp_interface モジュールは、ルータの疑似インターフェイスに関する情報と各 VRID が保持する情報を格納する。また、vrrp_event モジュールから受けた設定の変更に従い、疑似インターフェイスの設定変更を行う。

各疑似インターフェイスが保持する vrrp_interface 構造体を図 6.2に示す。

疑似インターフェイスの設定は、KAME によって新たに実装された ioctl 関数を利用する。

vrrp_interface_vrrif_set 関数は、ioctl 関数を利用し、疑似インターフェイスの物理インターフェイスと MAC アドレスを設定する。その中身を図 6.3に示す。

6.2.2　vrrp_packet モジュール

vrrp_packet モジュールは、ルータが送受信するパケットに関するデータを格納し、その送受信を行う。送受信するパケットには、VRRP 生存広告パケット、NDP 近隣広告パケットの二種類がある。

```
int
vrrp_interface_vrrif_set (char *if_name, u_int parent_index,
                          struct ether_addr *lladdr)
{
  int             sd;
  struct ifreq    ifr;
  struct vrrpreq  vrreq;

  bzero (&ifr, sizeof (ifr));
  bzero (&vrreq, sizeof (vrreq));

  sd = socket (AF_INET, SOCK_DGRAM, 0);
  if (sd == -1)
    {
      zlog_warn ("cannot open socket for changing "
                 "ip address of interface %s: %m", if_name);
      return -1;
    }
  strncpy (ifr.ifr_name, if_name, sizeof(ifr.ifr_name));
  ifr.ifr_data = (caddr_t)&vrreq;
  vrreq.vr_parent_index = parent_index;
  vrreq.vr_lladdr.sa_family = AF_LINK;
  vrreq.vr_lladdr.sa_len = ETHER_ADDR_LEN;
  bcopy (lladdr, vrreq.vr_lladdr.sa_data, ETHER_ADDR_LEN);

  if (ioctl (sd, SIOCSETVRRP, (caddr_t) &ifr) == -1)
    {
      zlog_warn ("cannot set vrrp parent interface %s "
                 "(ioctl): %m", if_name);
      close (sd);
      return -1;
    }

  close (sd);
  return 0;
}
```

図 6.3: vrrp_interface_vrrif_set() 関数

VRRP 生存広告パケットの送受信は、Raw ソケットによって行われる。マスタールータは、vrrp_state モジュールより受けとった現在の状態を元にパケットを作成し、送信する。バックアップルータは、マスターから受信したパケットを解析し、受け取るか破棄するかを判断する。受信したパケットの内容と現在の自分の状態から、状態の変更の判断を行う。変更する場合は vrrp_state モジュールへ移行する。

NDP 近隣広告パケットの送信は、Raw ソケットによって行われる。

6.2.3　vrrp_event モジュール

vrrp_event モジュールは、状態の変更に関する動作を行う。

具体的には、初期化状態の動作を行う state_initialize 関数、マスター状態への設定変更を行う change_to_master 関数、バックアップ状態への設定変更を行う change_to_backup 関数がある。

6.2.4　vrrp_zebra モジュール

図 6.4に vrrp_zebra モジュールの概念を示す。

vrrp_zebra モジュールは、Zebra 独自のプロトコルを用いて、zebra デーモンとシステム情報を交換する。交換されるシステム情報は、ルータインターフェイス状態、ルータ広告の送受信に関するもの、そして vrrpd の持つ経路情報である。

zebra デーモンからルータインターフェイスの状態が渡された場合、適切な vrrp_interface モジュールの状態を更新する。ルータインターフェイスの状態は、vrrp_zebra モジュールから zebra デーモンへ渡されることは無い。

vrrp_event モジュールで状態に変更があった場合、vrrp_zebra モジュールは zebra デーモンへとルータ広告の送受信に関する情報を送信する。ルータ広告の送受信に関する情報は次の二種類である

- 送信の有無
  マスター状態に変更が移行した時に、ルータ広告の送信を始める。それ以外の状態では、ルータ広告の送信を止める。
- 送信するインターフェイス
  ルータ広告を送信するインターフェイスを指定する。指定するインターフェイスは、vrrpd で IPv6 アドレスや MAC アドレスを設定した疑似インターフェイスである。それによって、終端ネットワーク内のホストのデフォルトゲートウェイを仮想ルータに向けることが可能になる。

また、vrrp_event モジュールで状態に変更があった場合、vrrp_zebra モジュールは zebra デーモンへ経路情報を送信する。マスター状態に移行した場合、予め設定しておいたプレフィックス情報を zebra デーモンに渡す。逆にマスター以外の状態になった場合、zebra デーモンからプレフィックス情報を消去する。

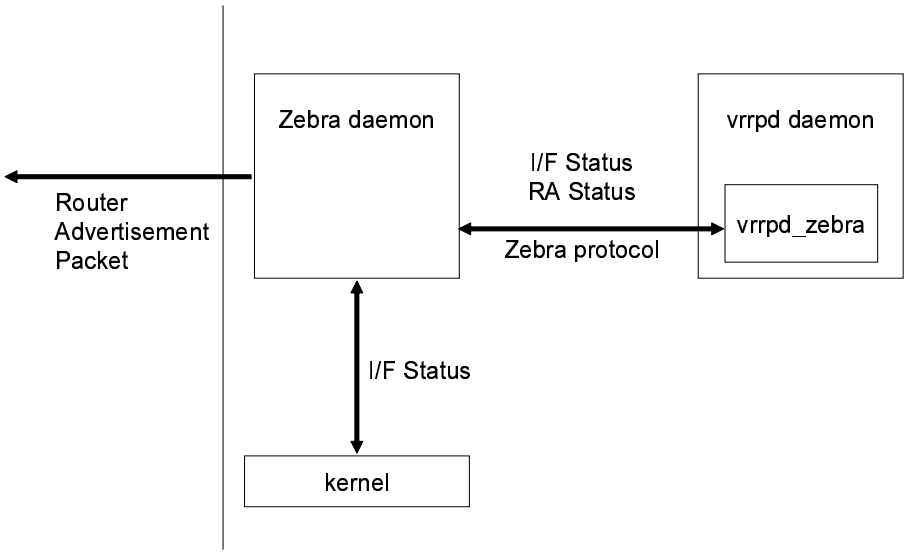

図 6.4: vrrp_zebra モジュールの概念

6.2.5　vrrp_top モジュール

vrrp_top モジュールは、VRRP デーモン全体の動作を制御するモジュールである。vrrp_interface モジュールなどで行った設定を用いた動作の開始、終了を行う。

6.3　vrrpd の動作概要

図 6.1を用いて、vrrpd の動作概要を説明する。

Zebra デーモンは vrrpd より前に起動され、カーネルとの間でインターフェイスやカーネルの情報を交換しておく (①)。vrrpd が起動するとまず、Zebra デーモンとの接続を開始し、インターフェイスの情報を入手する (②)。vrrp_top モジュールによって、VRRP の動作が開始する (③)。vrrp_event モジュールは、vrrp_zebra モジュールから入手したインターフェイスの情報 (④) や設定されたプライオリティから状態の変更を行う。そして、vrrp_event モジュールは状態によって、生存広告の送受信の制御 (⑤) や、疑似インターフェイスの設定変更を行う (⑥)。疑似インターフェイスの IP アドレスや MAC アドレスの変更は vrrp_interface モジュールがカーネルに対して行う (⑦)。その後は、vrrp_packet モジュールが受信したパケットの内容などから、vrrp_event モジュールが状態変更を行う。

6.4　vrrpd の使用方法

以下に、vrrpd の使用方法を示す。[] で囲まれた記述は、省略することが可能になっていることを示す。

6.4.1　起動

vrrpd は、設定ファイルを利用して動作する。デフォルトの設定ファイルは、/usr/local/etc/vrrpd.conf であるが、-f コマンドラインオプションによって変更可能である。

また、-d コマンドラインオプションによって、デーモンモードで動作する。デフォルトではフォアグラウンドのプロセスとなる。

-P コマンドラインオプションは、後述の動的設定ターミナルの待ち受け TCP ポート番号を指定するためのものである。デフォルトは 2607 である。

```
# vrrpd [-d] [-f 設定ファイル ] [-P ポート番号]
```

vrrpd は zebra daemon から インターフェース情報を取得する。そのため、 zebra daemon をあらかじめ起動しておく必要がある。

また、vrrpd は空の設定ファイルを最低限必要とする。設定ファイルが存在しないとき、vrrpd は起動されない。

6.4.2　動的設定ターミナル

動的設定ターミナルは、ユーザに設定や状態の確認、動的な設定の変更を提供する TELNET[22] ユーザインターフェースである。

```
% telnet ::1 2607
Trying ::1...
Connected to localhost.
Escape character is '^]'.

Hello, this is zebra (version 0.95-pre1).
Copyright 1996-2002 Kunihiro Ishiguro.


User Access Verification

Password:
```

動的設定ターミナルは、認証、アクセス制限をサポートしている。また、動的ターミナルの内部で動作するシェルは、コマンドラインの編集、コマンドヒストリ、コマンドの補完、ヘルプの表示など様々な機能を有している。

設定ファイルに記述できるコマンドは全て動的設定ターミナルを通じて可能である。また、動的設定ターミナルを通じて設定できるコマンドはほぼすべて設定ファイルにおいても可能である。例外として、動的設定ターミナルにしか存在しないコマンドは、show コマンド、enable コマンド、configure コマンドである。これらは、動的設定ターミナルのみに利用される目的を持つコマンドである。

ユーザは、動的ターミナルに接続し、認証された直後、VIEW NODE に位置する。VIEW NODE は、状態の確認のみを行なえる制限された NODE である。現在自分がどの NODE にいるかは、

ターミナルのシェルプロンプトによって識別することができる。VIEW NODE のプロンプトの最後は > となっている。

```
Hostname>
```

enable コマンドは、VIEW NODE から ENABLE NODE に移行するためのコマンドである。ENABLE NODE に移行することは、管理権限が許可されたことを意味する。VIEW NODE と同じ show コマンドを持ち、状態の確認が行なえる。また、VIEW NODE では許されていない設定ファイルの確認など、管理権限を持つ者に対してのみ許される show command も存在する。ENABLE NODE のプロンプトの最後は # となっている。

```
Hostname> enable
Password:
Hostname#
```

configure コマンドは、ENABLE NODE から CONFIG NODE に移行するためのコマンドである。CONFIG NODE 内では、ユーザは様々な設定を行なえる。configure terminal コマンドは、動的設定ターミナルから設定を行なう、という指定を意味する。CONFIG NODE では、プロンプトに (config) が挿入される。

```
Hostname# configure terminal
Hostname(config)#
```

現在の NODE を抜け、直前の NODE に戻る場合は exit コマンドを使用する。

```
Hostname(config)# exit
Hostname#
```

6.4.3　インターフェイスの設定

vrrpd の設定は、全て疑似インターフェイスにある vrrp_inteface 構造体に保存されている。そのため、各 VRID の設定は全てインターフェイスの設定としてなされる。インターフェイスの各種の設定を行なうためには、INTERFACE NODE を利用する。INTERFACE NODE への移行は、CONFIG NODE の interface コマンドによって行なえる。INTERFACE NODE では、プロンプトに (config-if) が挿入される。

```
Hostname(config)# interface IfName
Hostname(config-if)#
```

ここに示す設定は、vrrp0、vrrp1 などの疑似インターフェイスで意味のある設定である。そのため、それ以外のインターフェイスの場合は設定を許可しない。

設定可能な VRRP インターフェイスの値を、コマンド、デフォルト値とともに以下に説明する。

- ipv6 vrrp id

ipv6 vrrp id <1-255>

ipv6 vrrp id コマンドによって、この VRRP インターフェイスが保持する VRID を設定する。このコマンドによって所属する VRRP グループが決定する。デフォルト値は設定されておらず、必ず設定しなくてはならない。

- ipv6 vrrp physical-interface

ipv6 vrrp physical-interface *IfName*

ipv6 vrrp physical-interface コマンドによって、VRRP 疑似インターフェイスの通信を実際に行う物理インターフェイスを設定する。

- ipv6 vrrp address

ipv6 vrrp address *IPv6Address*

ipv6 vrrp address コマンドによって、疑似インターフェイスが所有する仮想 IPv6 アドレス値を設定する。ここで設定する IPv6 アドレスはその終端ネットワークのホストが設定するデフォルトゲートウェイのアドレスとなる。デフォルトゲートウェイのアドレスはリンクローカルアドレス [23] と定まっている。そのため、リンクローカルアドレスの規格に沿わないアドレスは設定不可能とする。

- ipv6 vrrp address owner

ipv6 vrrp address owner

ipv6 vrrp address owner コマンドは、そのルータの物理インターフェイスが持つアドレスを、VRRP グループで共有する事を設定する。
ipv6 vrrp physical-interface コマンドで設定された物理インターフェイスが持つリンクローカルアドレスを利用する。そのため、physical-interface コマンドを先に実行しなくてはならない。
仮想 IP アドレスを所有しているルータは、優先度値は 255 になる。そのため、このコマンドが実行された時に、保持する優先度値を 255 に変更する。

- ipv6 vrrp priority

ipv6 vrrp priority <1-254>

ipv6 vrrp address コマンドによって、そのルータが保持する優先度値を設定する。優先度値 255 は、アドレスを元々所有している場合のみに持てる値なので、このコマンドでは設定出来ない。ipv6 vrrp address owner コマンドのみで設定出来るようにする。
デフォルトの値は 100 である。

- ipv6 vrrp preempt

  ipv6 vrrp preempt

  ipv6 vrrp preempt コマンドによって、そのルータの Preempt モードを有効にする。デフォルトでは有効になっている。

- ipv6 vrrp address from-master

  ipv6 vrrp address from-master

  ipv6 vrrp addres from-master コマンドによって、マスターからの広告で保持するアドレスを上書きする機能を有効にする。デフォルトでは無効になっている。ipv6 vrrp address owner コマンドを実行している場合は、常にマスターであるはずなので、実行できない。

- ipv6 vrrp upstream-interface

  ipv6 vrrp upstream-interface *IfName*

  ipv6 vrrp upstream-interface コマンドによって、ルータの上流ネットワークに接続されているインターフェイスを指定する。このコマンドによって指定されたインターフェイスが無効になった時に VRRP の動作終了する。ルータの上流インターフェイスを監視し、障害判断を正確に行う事が可能になる。

- ipv6 vrrp prefix

  ipv6 vrrp prefix *Prefix*

  ipv6 vrrp prefix コマンドによって、vrrpd の下流ネットワークの prefix を指定する。vrrpd は、この Prefix の経路情報を Zebra に渡す。この経路情報は経路制御プロトコルが、再配布する時に利用される。また、vrrpd はマスター状態時のみ経路情報を渡す。それによって、経路制御プロトコルの向けるルータをマスタールータにすることが可能になる。

6.4.4　動作設定

インターフェイスの設定で作成した設定を、動作設定によって実際に動作させる。

```
Hostname(config)# router vrrp
Hostname(config-vrrp)# interface vrrp0
Hostname(config-vrrp)# no interface vrrp0
Hostname(config-vrrp)# show vrrp state
```

CONFIG NODE から、上のような操作を行い、動作を設定する。

router vrrp と入力することで VRRP NODE へと移行する。

VRRP NODE 内で、interface コマンドを入力すると、そのインターフェイスが動作を開始する。INTERFACE NODE での設定によって動作内容が決定するので、INTERFACE NODE での設定が不十分の場合は動作しない。必要最低限の設定は、VRID、優先度値、物理インターフェイスである。

no interface コマンドは interface コマンドによって有効にしたインターフェイスを無効にするコマンドである。

show ipv6 vrrp コマンドは、現在各インターフェイスに設定された情報を全て表示するコマンドである。次のように表示される。

```
Hostname(config-vrrp)# show ipv6 vrrp

VRRP Interface Name : vrrp0
  VRID                    : 1
  Physical Interface Name : ed1
  Share IP Address        : fe80::1:1:1:1
  Share MAC Address       : 0:0:5e:0:2:1
  Priority                : 200
  Preempt Mode            : TRUE
  Advertisement Interval  : 1
  Upstream Interface Name : ep0
  State                   : MASTER
    Advertisement Timer   : 00.43

VRRP Interface Name : vrrp1
  VRID                    : 0
  Physical Interface Name : (null)
  Share IP Address        : ::
  Share MAC Address       : 0:0:5e:0:2:2
  Priority                : 0
  Preempt Mode            : FALSE
  Advertisement Interval  : 1
  Upstream Interface Name : (null)
  State                   : DISABLE
```

vrrp0 は INTERFACE NODE で全ての設定を行っているので、interface コマンドによって、動作が開始している。現在の状態はマスターである。マスター状態の場合、次に広告を送信するまでの時間である、広告タイマーが表示される。

vrrp0 は設定が完了してないので、動作していない。その場合、状態は DISABLE と表記される。

これらを設定することによって、VRRP の動作に必要な設定を完了することが出来る。

第7章　本システムの評価

本章では、本研究で作成したシステムの評価として、想定動作環境で動作させ、評価を行った。

7.1　実験１:切り替え時間の測定

マスターに障害が発生してからバックアップルータがマスターに切り替わるまでにかかる時間を測定した。

7.1.1　実験の目的

本システムの切り替え時間を測定し、その値が Internet-Draft に準拠したものかを評価する。

バックアップルータは、3.3章の計算式によってマスター障害を検知するまでの時間を計算する。

また、広告インターバルはデフォルト値で 1 秒である。そのため、障害発生のタイミングによって、理論値で一秒の差が生じる。

本実験は、理論値と実測値を比較し、本システムの動作が Internet-Draft に準拠していることを確認した。

7.1.2　実験方法

ntp を用い各ルータのタイマーを同期した。

マスタールータの vrrpd を終了し、その時間を gettimeofday 関数によって記録した。そしてバックアップルータがマスターの障害を検知し、マスター状態へ推移した時間を記録した。この時間の差を計算したものを切り替え時間とした。

7.1.3　実験結果

バックアップルータの優先度値、1、50、100、150、200、250 それぞれにおいて、10 回計測を行った。表 7.1と図 7.1にその結果をまとめた。

図 7.1の上下の直線は理論値の最大値、最小値である。本実験の結果、切替時間は理論値の範囲に収まっている。

以上の結果、各優先度値において、その理論と比べ適切な間隔で切替が行われていることが分かった。

	優先度値	1	50	100	150	200	250
理論値	最大値	3.996	3.805	3.609	3.414	3.219	3.023
	最小値	2.996	2.805	2.609	2.414	2.219	2.023
	平均値	3.496	3.305	3.109	2.914	2.719	2.523
実測値	最大値	3.829	3.764	3.534	3.371	3.041	2.991
	最小値	3.095	2.876	2.830	2.597	2.535	2.098
	平均値	3.442	3.286	3.164	2.825	2.798	2.501

表 7.1: 切替時間

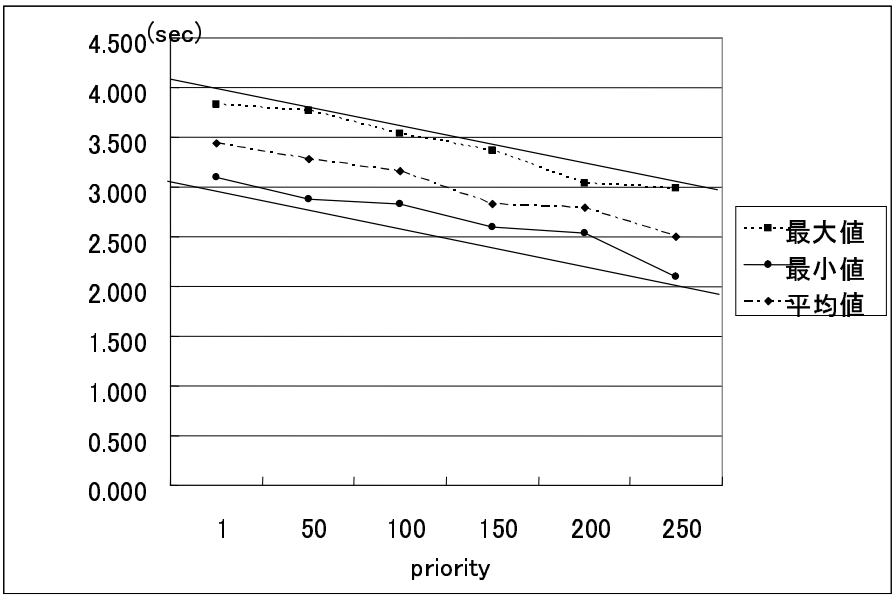

図 7.1: 切替時間

7.2　実験２:VRRP 単独動作

ルータの上下に VRRP を動作させる環境での評価を行った。

7.2.1　実験の目的

ルータの上下でそれぞれ VRRP を動作させるという環境で、VRRP デーモンの動作を確認する。

7.2.2　実験環境

環境の詳細を次に示す。

図 7.2に示す実験環境を作り、本システムを動作させた。vrrp1、vrrp2 の２台の FreeBSD マシンを用意し、それぞれのルータに VRRPD を追加した Zebra パッケージをインストールした。本研究の環境を表 7.2にまとめた。

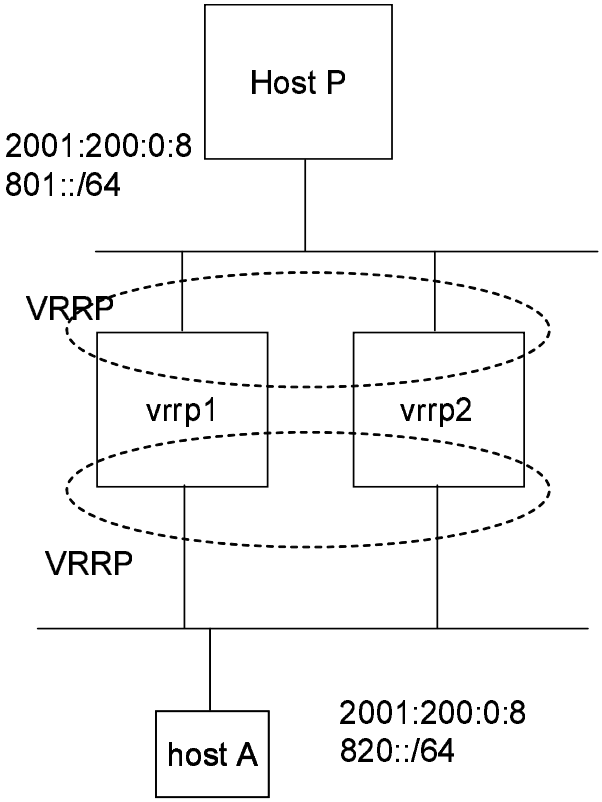

図 7.2: VRRP 単独実験環境

- 利用した IP アドレス

  ２台のルータの上流側のネットワークには 2001:200:0:8801::/64 のアドレスが割り当てられている。また、下流側のネットワークには 2001:200:0:8820::/64 のアドレスが割り当てられている。それぞれのルータには表 7.3に示すアドレスを設定した。

- 各ホストの経路表

  HOST P の経路表に、2001:200:0:8820::/64 向けの経路を上流側 VRRP の仮想 IP アドレスに向ける経路を追加した。HOST A は vrrp1 から受けたルータ広告によって下流側 VRRP の仮想 IP アドレスにデフォルトの経路を向ける。

	vrrp1	vrrp2
上流側 I/F アドレス	2001:200:0:8801::43:1	2001:200:0:8801::43:2
下流側 I/F アドレス	2001:200:0:8820::1	2001:200:0:8820::2
上流側 VRRP Priority	200	100
下流側 VRRP Priority	200	100

表 7.2: VRRP 単独実験各ルータの設定

Zebra では、ルータ広告によって広告するプレフィックスを指定した。vrrp1 の Zebra の設定ファイルを図 7.3に示す。

マスタールータの vrrpd の設定を図 7.4に示す。

7.2.3　実験内容

1. HOST P から HOST A へストリーミング映像を流した。
2. 正常状態でマスターである vrrp1 の上下の回線を引き抜いた。
3. HOST A で、ストリーミングの切断時間を測定した。
4. 経路回復後、回線を挿し直した。

以上の実験を行っている間、マスター、バックアップそれぞれのルータでパケットの送受信数を毎秒観測した。これによって実際に利用されるルータを判断した。

7.2.4　実験結果

各ルータのパケット送受信数を観察した結果、回線切断時、往路・復路それぞれにおいて経路が vrrp2 を通るものに変更された。逆に、回線の再接続を行った場合経路は vrrp1 へと再変更された。切替時間は、10 回計測を行った平均値が 3.30 秒となった。

```
!
! Zebra configuration saved from vty
!   2003/12/27 18:32:35
!
hostname vrrp1
password zebra
!
interface pfsync0
 ipv6 nd suppress-ra
!
interface lo0
!
interface ppp0
 ipv6 nd suppress-ra
!
interface sl0
 ipv6 nd suppress-ra
!
interface vrrp0
 no ipv6 nd suppress-ra
 ipv6 nd prefix-advertisement 2001:200:0:8820::/64 2592000 604800 onlink autoconfig
!
interface vrrp1
 ipv6 nd suppress-ra
!
interface faith0
 ipv6 nd suppress-ra
!
interface gif0
 ipv6 nd suppress-ra
!
interface pflog0
 ipv6 nd suppress-ra
!
interface ep0
 ipv6 nd suppress-ra
!
interface ed1
 ipv6 nd suppress-ra
!
!
line vty
!
```

図 7.3: 実験 2 :zebra の設定ファイル

```
!
! Zebra configuration saved from vty
!   2004/01/05 14:41:28
!
hostname vrrp1
password zebra
!
interface vrrp0
  ipv6 vrrp id 1
  ipv6 vrrp physical-interface ed1
  ipv6 vrrp address fe80::1:1:1:1
  ipv6 vrrp priority 254
  ipv6 vrrp preempt
  no ipv6 vrrp address from-master
  ipv6 vrrp prefix 2001:200:0:8820::/64
!
interface vrrp1
  ipv6 vrrp id 2
  ipv6 vrrp physical-interface ep0
  ipv6 vrrp address fe80::1:1:1:2
  ipv6 vrrp priority 254
  ipv6 vrrp preempt
  no ipv6 vrrp address from-master
  ipv6 vrrp prefix 2001:200:0:8801::/64
!
router vrrp
  interface vrrp0
  interface vrrp1
!
line vty
!
```

図 7.4: 実験 2 :vrrpd の設定ファイル

7.3　実験３:OSPF と VRRP の協調

ルータの上流では OPSF、下流では VRRP を動作させた環境で実験を行った。

7.3.1　実験の目的

本研究で実装した vrrpd と ospf6d の連携を行う。そして、様々な箇所に障害を発生させ経路の収束を確認する。それによって、対応可能な障害とそうでないものを切り分ける。

7.3.2　実験環境

環境の詳細を次に示す。

図 7.5に示す実験環境を作り、本システムを動作させた。vrrp1、vrrp2 の２台の FreeBSD マシンを用意し、それぞれのルータに VRRPD を追加した Zebra パッケージをインストールした。

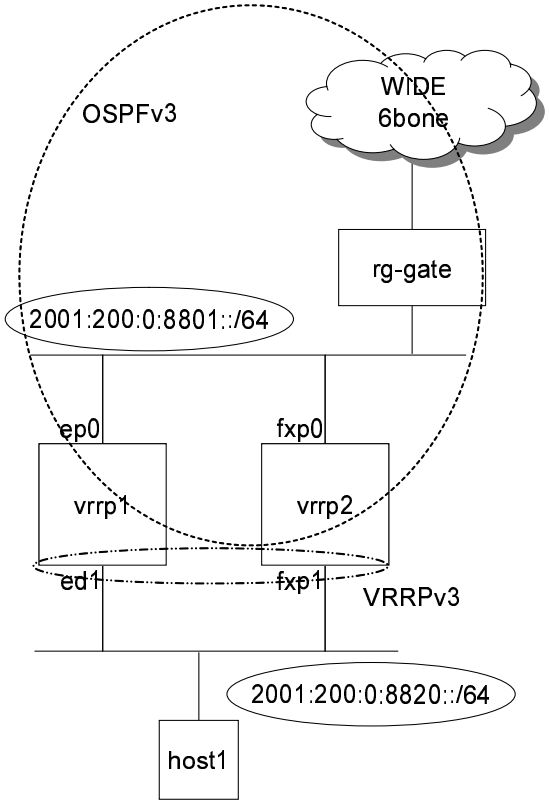

図 7.5: OSPF と VRRP の協調実験環境

- 利用した IP アドレス

  ２台のルータの上流側のネットワークには 2001:200:0:8801::/64 のアドレスが割り当てられている。また、下流側のネットワークには 2001:200:0:8820::/64 のアドレスが割り当てられている。それぞれのルータには表 7.3に示すアドレスを設定した。

	vrrp1	vrrp2
上流側 I/F アドレス	2001:200:0:8801::43:1	2001:200:0:8801::43:2
下流側 I/F アドレス	2001:200:0:8820::1	2001:200:0:8820::2
VRRP Priority	200	100
OSPF 再配布 Metric	20000	10000

表 7.3: OSPF と VRRP の協調実験環境の各ルータの設定

- zebra の設定

  各 zebra デーモンに、ルータ広告によって広告するプレフィックスの設定を行う。本実験では、2001:200:0:8820::/64 を広告する。

  実際の vrrp1 の設定ファイルを図 7.8に示す。

- vrrpd の設定

  三台のルータの下流側のインターフェイスを利用して VRRP の設定を行った。共有するリンクローカルアドレスは fe80::12 を利用した。それぞれのルータの優先度値は、vrrp1 は 200、vrrp2 は 100 と設定した。

  実際の vrrp1 の設定ファイルを図 7.9に示す。

- ospf6d の設定

  OSPF は Zebra パッケージから、ospf6d を利用した。

  各ルータでは上下それぞれのインターフェイスを OSPF によって監視する。上流のインターフェイスには最小のコスト値である 1 を設定した。逆に下流のインターフェイスには 60000 と設定した。

  vrrpd から送られる経路の再配布を行った。再配布を行う経路は、下流インターフェイスの接続経路と同じものである。そのため、通常状態では接続経路が優先され、再配布による経路の切り替えが出来ない。そこで、本実験では接続経路の広告を拒否し、再配布経路を有効化した。

  各ルータの設定で変更を加えた点は、VRRP からの再配布メトリック (metric) である。この値を、VRRP の優先度が低い方を少ない値にする。vrrp1 は 20000、vrrp2 は 10000 と設定した。

  実際の vrrp1 の設定ファイルを図 7.10に示す。

7.3.3　実験方法

WIDE 6bone 内のサーバーから HOST1 へストリーミング映像を流した。この環境で次に示す障害を発生させ、その結果を観察した。また、回線収束後元の状態に戻し、その結果も観察した。観察方法は、通信の切断時間の測定、各インターフェイスの送受信パケット量の観測、によって行った。

1. マスタールータの上流の障害

   vrrp1 の上流のインターフェイスである ep0 から回線を抜いた。

2. マスタールータ本体の障害

   vrrp1 の本体に障害を起こした。本実験では、vrrp1 の上下のインターフェイスから回線を抜くことで、本体に障害が発生した状況を作り出した。

3. マスタールータの下流の障害

   vrrp1 の下流のインターフェイスである ed1 から回線を抜いた。

7.3.4　実験結果

本実験の結果は次のようになった。以下に、各障害毎の動作結果を示す。

1. マスタールータ上流の障害

   マスタールータ上流に障害が発生すると、VRRP は障害を検知できないため経路変更を行わなかった。

   OSPF は回線の切断を判断し、経路の再計算を行った。再計算の終了後、host1 から rg-gate までの経路は、「host1 → vrrp1(ed1) → vrrp1(ed1) → vrrp2(fxp1) → vrrp2(fxp0) → rg-gate」という経路に収束した。

   回線切断から通信の回復までの時間を測定した。10 回測定した平均値は 31.8 秒となった。この値は、OSPF の再計算に必要な時間と考えられる。

   経路切断から復活までを図 7.6に示す。

2. マスタールータ本体の障害

   まず、vrrp1 からの広告が届かなくなるため、vrrp2 が障害を判断しマスターへと推移した。vrrp1、vrrp2 それぞれのルータがマスターになるため両ルータから OSPF への経路再配布が行われる。OSPF では vrrp1 への経路がなくなるため、vrrp2 から再配布された経路が採用され、帰りの経路が vrrp2 へと変更される。よって往路復路共に vrrp2 を通る経路へと収束した。

   回線切断から通信の回復までの時間測定した。10 回測定した平均値は 3.22 秒となった。この値は、VRRP の経路変更に依存している。

3. マスタールータ下流の障害

   まず、vrrp1 からの広告が届かなくなるため、vrrp2 が障害を判断しマスターへと推移した。vrrp1、vrrp2 それぞれのルータがマスターになるため両ルータから OSPF への経路再配布が行われる。vrrp2 で指定した再配布メトリック値が vrrp1 より低いため、帰りの経路が vrrp2 へと変更される。よって往路復路共に vrrp2 を通る経路へと収束した。

   回線切断から通信の回復までの時間測定した。10 回測定した平均値は 3.15 秒となった。この値は、VRRP の経路変更に依存している。

   経路切断から復活までを図 7.7に示す。

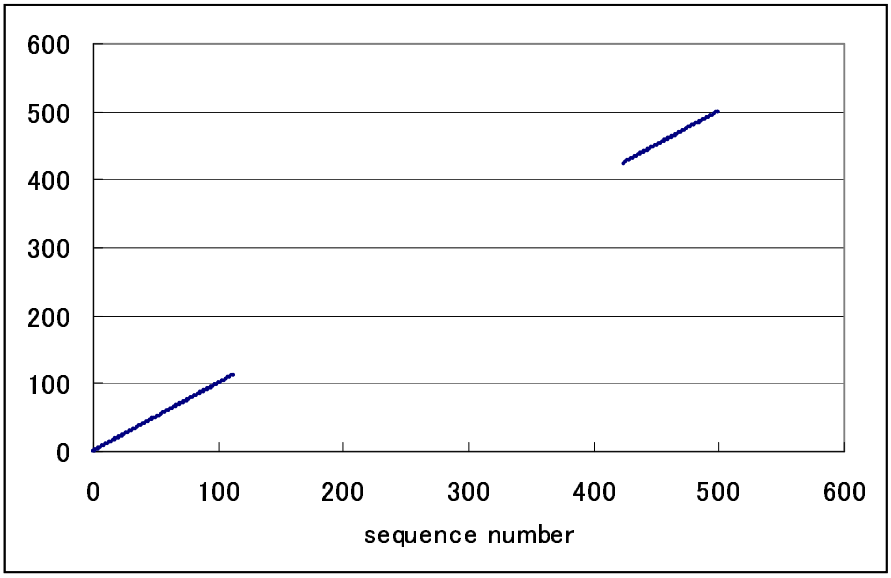

図 7.6: 実験 3 :マスタールータ上流障害時の切断状況

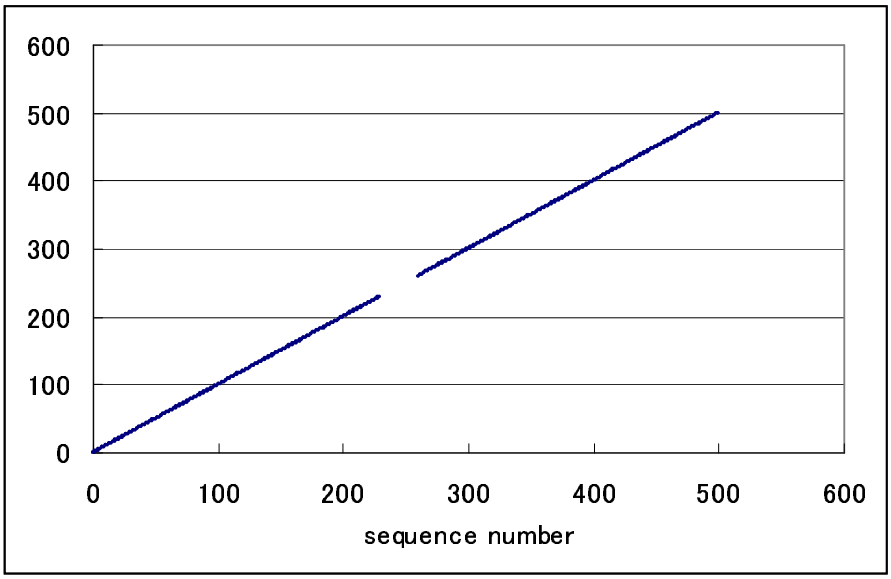

図 7.7: 実験 3 :マスタールータ下流障害時の切断状況

```
!
! Zebra configuration saved from vty
!   2003/12/27 18:32:35
!
hostname vrrp1
password zebra
!
interface pfsync0
 ipv6 nd suppress-ra
!
interface lo0
!
interface ppp0
 ipv6 nd suppress-ra
!
interface sl0
 ipv6 nd suppress-ra
!
interface vrrp0
 no ipv6 nd suppress-ra
 ipv6 nd prefix-advertisement 2001:200:0:8820::/64 2592000 604800 onlink autoconfig
!
interface vrrp1
 ipv6 nd suppress-ra
!
interface faith0
 ipv6 nd suppress-ra
!
interface gif0
 ipv6 nd suppress-ra
!
interface pflog0
 ipv6 nd suppress-ra
!
interface ep0
 ipv6 nd suppress-ra
!
interface ed1
 ipv6 nd suppress-ra
!
!
line vty
!
```

図 7.8: 実験 3 :zebra の設定ファイル

```
!
! Zebra configuration saved from vty
!   2004/01/05 14:41:28
!
hostname vrrp1
password zebra
!
interface vrrp0
  ipv6 vrrp id 1
  ipv6 vrrp physical-interface ed1
  ipv6 vrrp address fe80::1:1:1:1
  ipv6 vrrp priority 254
  ipv6 vrrp preempt
  no ipv6 vrrp address from-master
  ipv6 vrrp prefix 2001:200:0:8820::/64
!
router vrrp
  interface vrrp0
!
line vty
!
```

図 7.9: 実験 3 :vrrpd の設定ファイル

```
!
! Zebra configuration saved from vty
!   2004/01/14 22:45:44
!
hostname ospf6d@vrrp1
password zebra
log file /var/log/zebra-ospf6d.log
!
!
interface ep0
 ipv6 ospf6 cost 1
 ipv6 ospf6 hello-interval 10
 ipv6 ospf6 dead-interval 40
 ipv6 ospf6 retransmit-interval 5
 ipv6 ospf6 priority 0
 ipv6 ospf6 transmit-delay 1
 ipv6 ospf6 instance-id 0
!
interface ed1
 ipv6 ospf6 cost 60000
 ipv6 ospf6 hello-interval 10
 ipv6 ospf6 dead-interval 40
 ipv6 ospf6 retransmit-interval 5
 ipv6 ospf6 priority 1
 ipv6 ospf6 transmit-delay 1
 ipv6 ospf6 instance-id 0
 ipv6 ospf6 advertise prefix-list connected-deny
!
router ospf6
 router-id 0.0.43.1
 redistribute vrrp route-map connected-vrrp
 interface ep0 area 0.0.0.0
 interface ed1 area 0.0.0.0
!
access-list access4 permit 127.0.0.1/32
!
ipv6 access-list access6 permit ::1/128
!
ipv6 prefix-list connected-deny seq 5 deny 2001:200:0:8820::/64
ipv6 prefix-list vrrp-segment seq 5 permit 2001:200:0:8820::/64
ipv6 prefix-list vrrp-segment seq 10 deny any
!
route-map connected-vrrp permit 100
 match ipv6 address prefix-list vrrp-segment
 set metric-type type-2
 set metric 20000
!
line vty
 access-class access4
 ipv6 access-class access6
!
```

図 7.10: 実験 3 :ospf6d 設定ファイル

第8章　結論

8.1　まとめ

本研究では、インターネットにおける終端ネットワークの問題点を解決するシステムを構築した。

現状の終端ネットワークは、ゲートウェイルータがシングルポイント障害となってしまう問題点があった。本研究では、VRRP を用い上下流の経路制御システムとの連携を行う行う方法によってこの問題を解決した。

本研究では、Zebra パッケージ上で VRRP を実装した。また、実用的なネットワーク構成の例を示し、他の経路制御システムとの連携を考えた運用方法を示した。そして、本研究によって VRRP を追加した Zebra パッケージを用い、ネットワークを構築しその評価を行った。

本研究の成果により、ユーザの利用する終端ネットワークの信頼性は飛躍的に向上した。これにより、従来に比べ、信頼性を必要とするアプリケーションの利用も可能になる。

8.2　今後の課題

本研究で実装したシステムは実験環境による評価を行った。しかし、本システムを実際にユーザが利用しているネットワークにて運用することによって、本システムの有益性を検証する必要がある。

また、本研究によって、ルータに発生した障害に関する問題は解決できた。しかし、本研究によって解決できない問題に、途中回線の障害などによるセグメント分割がある。本研究は、終端ネットワーク上の経路制御に関係する様々な問題点に対応し、ユーザの利用環境を向上を行う必要がある。

謝辞

本研究をすすめるにあたり、ご指導頂きました慶應義塾大学環境情報学部教授 村井純博士、並びに政策・メディア研究科 徳田英幸博士、環境情報学部助教授 楠本博之博士、同学部助教授 中村修博士に感謝いたします。また、日頃より研究活動のご指導をいただきました、慶應義塾大学環境情報学部専任講師 南政樹氏に深く感謝いたします。

絶えず研究に対する指導をしていただいた、政策・メディア研究科 小原泰弘氏には特に感謝いたします。クリスマスイブの夜、夢に出て叱咤激励していただいたご恩は忘れません。

本論文の作成に対するアドバイスを頂いた、政策メディア・研究科 岡田耕司氏、小柴晋氏に感謝いたします。また、修士論文に忙しい中、多くのアドバイスを頂いた、政策メディア・研究科 日野哲志氏、三屋光史朗氏に感謝いたします。

本研究を進めていく上でご支援下さった慶應義塾大学 徳田・村井・楠本・中村・南合同研究室 SING 研究グループの皆様に感謝の念を表します。

最後に、本論文に向け苦楽を共にした、高橋宏明氏、谷岡洋平氏、橋本和樹氏、成瀬大亮氏、廣瀬峻氏、金子紘子氏、白畑真氏、にゃんた氏をはじめとする同期の仲間達に感謝いたします。

参考文献

[1] J. Postel. *Internet Protocol*, September 1981. RFC 791.

[2] S. Deering and R. Hinden. *Internet Protocol, Version 6 (IPv6) Specification*, December 1998. RFC 2460.

[3] F. Baker and Ed. *Requirements for IP Version 4 Routers*, June 1995. RFC 1812.

[4] S. Knight, D. Weaver, D. Whipple, R. Hinden, D. Mitzel, P. Hunt, P. Higginson, M. Shand, and A. Lindem. *Virtual Router Redundancy Protocol*, April 1998. RFC 2338.

[5] T. Narten, E. Nordmark, and W. Simpson. *Neighbor Discovery for IP Version 6 (IPv6)*, December 1998. RFC 2461.

[6] K. Lougheed and Y. Rekhter. *Border Gateway Protocol (BGP)*, June 1990. RFC 1163.

[7] G. Malkin. *RIP Version 2*, November 1998. RFC 2453.

[8] J. Moy. *OSPF Version 2*, April 1998. RFC 2328.

[9] IEEE. http://standards.ieee.org/.

[10] R. Droms. *Dynamic Host Configuration Protocol*, March 1997. RFC 2131.

[11] Cisco systems, inc. http://www.cisco.com/.

[12] T. Li, B. Cole, P. Morton, and D. Li. *Cisco Hot Standby Router Protocol (HSRP)*, March 1998. RFC 2281.

[13] R. Hinden/Nokia. *Virtual Router Redundancy Protocol for IPv6*, June 2003. Work in Progress, draft-ietf-vrrp-ipv6-spec-05.txt.

[14] R. Hinden and S. Deering. *IP Version 6 Addressing Architecture*, July 1998. RFC 2373.

[15] Brian Haberman Steve Deering, Brian Zill. *IPv6 Scoped Address Architecture*, December 2003. draft-ietf-ipv6-scoping-arch-00.txt.

[16] 'internet engineering task fource'. http://www.ietf.org/.

[17] A. Conta and S. Deering. *Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification*, December 1998. RFC 2463.

[18] Gnu zebra. http://www.zebra.org/.

[19] Gnu general public license. http://www.gnu.org/copyleft/gpl.html.

[20] American national standards institute. http://www.ansi.org/.

[21] "the freebsd project". http://www.freebsd.org.

[22] J. Postel and J.K. Reynolds. Telnet protocol specification. Request For Comments 854, IETF, May 1983.

[23] R. Hinden and S. Deering. *Internet Protocol Version 6 (IPv6) Addressing Architecture*, April 2003. RFC 3513.