ルーターまたはロードバランサとしての Oracle® Solaris 11.2 システムの構成

Part No: E53806-02
2014 年 9 月

ORACLE®

目次

このドキュメントの使用法

- **概要** – Oracle Solaris 11.2 を IPv4 または IPv6 ルーターとして構成する方法について説明します。仮想ルーター冗長プロトコル (VRRP) および統合ロードバランサ (ILB) の概要および構成手順について説明します。
- **対象読者** – システム管理者。
- **前提知識** – ネットワーク管理に関する基本技術および一部の高度な技術。

製品ドキュメントライブラリ

この製品の最新情報や既知の問題は、ドキュメントライブラリ (http://www.oracle.com/pls/topic/lookup?ctx=E36784) に含まれています。

Oracle サポートへのアクセス

Oracle のお客様は、My Oracle Support を通じて電子的なサポートを利用することができます。詳細は、http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info (聴覚に障害をお持ちの場合は http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs) を参照してください。

フィードバック

このドキュメントに関するフィードバックを http://www.oracle.com/goto/docfeedback からお聞かせください。

♦ ♦ ♦ 第 1 章

1 ルーターおよびロードバランサの概要

この章では、コンピュータネットワークの接続および負荷分散にルーターとロードバランサが Oracle Solaris でどのように使用されるかについて説明します。ルーターは、ルーティング情報プロトコル (RIP)、次世代 RIP (RIPng)、Internet Control Message Protocol Router Discovery (RDISC)、Open Shortest Path First (OSPF)、ボーダーゲートウェイプロトコル (BGP)、Intermediate System to Intermediate System (IS-IS)、仮想ルーター冗長プロトコル (VRRP) などのプロトコルを使用して、ルーティングアクティビティーを制御します。

ロードバランサは、複数のサーバー間でネットワークトラフィックを分散します。ネットワークのワークロードの分散により、最適なリソース共有が達成され、スループットと可用性が向上します。

この章の内容は、次のとおりです。

ルーターの概要

ルーターは、コンピュータの接続およびネットワークのコンピュータ間のデータのパケット転送にコンピュータネットワークで使用されるデバイスです。ルーターは、さまざまなネットワークからの 2 つ以上の接続が可能です。ルーターは、受信データパケットからアドレス情報を読み取り、宛先を決定します。次に、パケットはルーターのルーティングテーブルの情報を使用して、次のネットワークに転送されます。このトラフィックはルーターのプロセスを指示し、データパケットが送信先ノードに到達するまで繰り返されます。

ルーティングプロトコル

ルーティングプロトコルは、システムのルーティングアクティビティーを処理します。ルーターは、ほかのホストとルーティング情報を交換し、リモートネットワークとの既知のルートを保持します。経路制御プロトコルはルーターとホストの両方で実行できます。ホストの経路制御プロトコルは、ほかのルーターやホストの経路制御デーモンと通信します。ホストはこれらのプロトコルを利用して、パケットの転送先を決定します。ネットワークインタフェースが使用可能な場合、システムは経路制御デーモンとの通信を自動的に行います。これらのデーモンは、ネットワーク上のルーターをモニターし、ローカルネットワーク上のホストにルーターのアドレスを通知します。すべてではありませんが、一部のルーティングプロトコルは統計情報も保持し、この統計を使って、ルーティングのパフォーマンスを計測できます。パケット転送と同様に、Oracle Solaris システム上でルーティングを明示的に構成する必要があります。

RIP と RDISC は標準 TCP/IP プロトコルです。次の表は、Oracle Solaris でサポートされているルーティングプロトコルの説明です。

表 1-1 Oracle Solaris ルーティングプロトコル

プロトコル	関連するデーモン	説明	手順の参照先
RIP	`in.routed`	IPv4 パケットをルーティングし、ルーティングテーブルを保持するインテリアゲートウェイプロトコル (IGP)	17 ページの「IPv4 ルーターの構成」
RDISC	`in.routed`	ホストがネットワーク上のルーターの存在を検索できるようにします。	『Oracle Solaris 11.2 でのネットワークコンポーネントの構成と管理 』の「単一インタフェースシステムのルーティングの有効化」
RIPng	`in.ripngd`	IPv6 パケットのルーティングおよびルーティングテーブルの維持を行う IGP	23 ページの「IPv6 対応のルーターを構成する方法」
近傍検索プロトコル (NDP)	`in.ndpd`	IPv6 ルーターの存在を通知し、ネットワーク上の IPv6 ホストの存在を検索します	『Oracle Solaris 11.2 でのネットワークコンポーネントの構成と管理 』の「IPv6 用にシステムを構成する方法」

Oracle Solaris のルーティングテーブルおよびタイプの詳細は、『Oracle Solaris 11.2 でのネットワークコンポーネントの構成と管理 』の「ルーティングテーブルとルーティングの種類」を参照してください。

ルーティング情報プロトコル

ルーティング情報プロトコル (RIP) は、距離ベクトル型ルーティングプロトコルです。RIP は、ルーティングメトリックとしてホップカウンタを使用します。ルーティングデーモン `in.routed` によって実装されます。デーモンは、システムがブートすると自動的に開始されます。`-s` オプションを指定してルーターで実行すると、`in.routed` デーモンは、到達可能なすべてのネットワークへのルートをカーネルルーティングテーブルに入力し、すべてのネットワークインタフェースを経由して到達可能性を通知します。`-q` オプションを指定して、ホストで実行すると、`in.routed` デーモンはルーティング情報を抽出しますが、到達可能性を通知しません。

ホストでは、ルーティング情報は次の 2 つの方法で抽出できます。

- フラグ (大文字の `S` または省スペースモード) を指定*しない*。`in.routed` デーモンは、ルーターと完全に同じように完全なルーティングテーブルを構築します。
- フラグを指定する。`in.routed` デーモンは、使用可能なルーターごとにデフォルトのルートを 1 つずつ含む最小カーネルテーブルを作成します。

ICMP ルーター発見プロトコル

ホストはルーター発見 (RDISC) プロトコルを使用して、ルーティング情報をルーターから取得します。ホストが RDISC を実行しているとき、ルーターは、ルーター情報の交換のために、RIP などの別のプロトコルも実行している必要があります。

RDISC は、ルーターおよびホストで実行する必要のある `in.routed` デーモンによって実装されます。ホストでは、`in.routed` は RDISC を使用して、RDISC によってアドレスを通知するルーターからデフォルトのルートを検出します。`in.routed` は、ルーターで RDISC を使用して、直接接続されているネットワーク上のホストにデフォルトのルートを通知します。詳細は、`in.routed`(1M) および `gateways`(4) のマニュアルページを参照してください。

Quagga ルーティングプロトコルスイート

Quagga は、Oracle Solaris を含む UNIX プラットフォームで RIP、RIPng、Open Shortest Path First (OSPF)、Intermediate System to Intermediate System (IS-IS)、およびボーダーゲートウェイプロトコル (BGP) のプロトコルの実装を可能にするルーティングソフトウェアスイートです。

RIPng は、IPv6 のさまざまな拡張機能を含む、IPv6 のサポートに RIP 拡張を提供します。RIPng の機能は RIP の機能に似ています。

OSPF は、大規模な自律システムネットワーク内にルーティング情報を分散するのに使用されるルータープロトコルです。OSPF の最新バージョンは OSPFv3 で、IPv6 のサポートが追加されています。

IS-IS は、大規模なサービスプロバイダネットワーク内のルーティング情報を分散するのに使用されるリンク状態動的ルーティングプロトコルです。

BGP は、IP ネットワークの接頭辞のセットを使用して、大規模な自律システムネットワーク間のパスと規則に基づいて、ルーティングを決定します。

次の表に、Oracle Solaris でサポートされているオープンソースの Quagga ルーティングプロトコルを一覧表示します。

表 1-2 Quagga ルーティングプロトコルスイート

プロトコル	関連するデーモン	説明
RIP	`ripd`	IPv4 パケットをルーティングし、ルーティングテーブルを近傍に通知する IPv4 距離ベクトル型 IGP
RIPng	`ripngd`	IPv6 パケットをルーティングし、ルーティングテーブルを保持する IPv6 距離ベクトル型 IGP
OSPF	`ospfd`	パケットのルーティングおよび高可用性ネットワークのための IPv4 リンク状態型 IGP。
BGP	`bgpd`	管理ドメイン間のルーティングのための IPv4 および IPv6 エクステリアゲートウェイプロトコル (EGP)
IS-IS	`isisd`	管理ドメインまたはネットワーク内のルーティングのための IPv4 および IPv6 リンク状態 IGP

Quagga プロトコルの詳細については、Quagga Routing Suite の Webサイト http://www.nongnu.org/quagga/index.html を参照してください。

仮想ルーター冗長プロトコル

VRRP は、ルーターやロードバランサに使用される IP アドレスなど、IP アドレスの高可用性を提供します。VRRP は、RFC 5798, Virtual Router Redundancy Protocol Version 3 for IPv4 and IPv6 に指定されているインターネット標準プロトコルです。Oracle Solaris では、VRRP サービスを構成および管理する管理ツールが用意されています。

Oracle Solaris 11.2 では、既存の標準レイヤー 2 VRRP に加えて、IPMP および InfiniBand インタフェース経由の VRRP のサポートおよびゾーン内の VRRP のサポート強化のために、独自のレイヤー 3 VRRP を提供しています。

VRRP の使用および VRRP ルーターの構成の詳細は、第3章「仮想ルーター冗長プロトコルの使用」および第4章「仮想ルーター冗長プロトコルの構成および管理」を参照してください。

VRRP ルーターの概要

仮想ルーター冗長プロトコルルーターは、VRRP を使用する 1 つ以上のルーターの動作によって作成される 1 つのルーターイメージです。VRRP は各 VRRP ルーター上で実行され、そのルーターの状態を管理します。1 つのホストで複数の構成された VRRP ルーターを持つことができ、各 VRRP ルーターは異なる仮想ルーターに属することができます。

レイヤー 2 VRRP ルーターは、標準の VRRP プロトコルを使用し、一意の仮想ルーター MAC アドレスが必要です。仮想 IP アドレスは、常に同じ仮想 MAC アドレスに解決されます。一意の仮想ルーター MAC アドレスを取得するには、VRRP VNIC を作成する必要があります。Oracle Solaris の独自のレイヤー 3 VRRP 機能では、VRRP ルーターの一意の VRRP 仮想 MAC アドレスの構成が完全に不要なため、IPMP および InfiniBand インタフェース経由の VRRP のサポートが提供されます。

VRRP の使用および VRRP ルーターの構成の詳細は、第3章「仮想ルーター冗長プロトコルの使用」および第4章「仮想ルーター冗長プロトコルの構成および管理」を参照してください。

統合ロードバランサの概要

Oracle Solaris では、統合ロードバランサ (ILB) でレイヤー 3 およびレイヤー 4 の負荷分散機能が提供されます。ILB は、SPARC ベースおよび x86 ベースのシステムにインストールされている Oracle Solaris オペレーティングシステムのネットワーク (IP) およびトランスポート (TCP/UDP) レイヤーで動作します。ILB は、信頼性とスケーラビリティーを向上し、ネットワークサービスの応答時間を最小限に抑えるために使用されます。

ILB はクライアントからの受信リクエストを傍受し、リクエストを処理するバックエンドサーバーを負荷分散規則に基づいて決定し、選択されたサーバーにリクエストを転送します。ILB は、バックエンドサーバーのルーターとしても使用できます。ILB はオプションの健全性検査を実行し、選

択されたサーバーが受信リクエストを処理できるかどうかを確認するために負荷分散アルゴリズムのデータを提供します。

ILB の機能

ILB の主な機能は次のとおりです。

- IPv4 および IPv6 について、ステートレス Direct Server Return (DSR) およびネットワークアドレス変換 (NAT) の動作モードをサポートします。

  DSR および NAT の動作モードについては、52 ページの「ILB の動作モード」を参照してください。
- 2 つの動作モードのアルゴリズムのセットを使用して、トラフィック、負荷分散、およびサーバー選択を支援します。
- コマンド行インタフェース (CLI) による ILB 管理を可能にします。

  CLI を使用した ILB の構成については、60 ページの「コマンド行インタフェースを使用した ILB の構成」を参照してください。
- 健全性検査によるサーバーモニタリング機能を提供します。

  サーバーモニタリング機能については、66 ページの「ILB の健全性検査のモニタリング」を参照してください。

次の表は、さまざまな動作モードで使用可能な ILB の機能のリストと説明です。

表 1-3 ILB の機能

機能	説明	動作のモード
クライアントが仮想 IP (VIP) アドレスを ping できるようにします。	ILB がクライアントから VIP アドレスへの ICMP エコーリクエストに応答します。	DSR および NAT の両方のモード
サービスを中断せずに、サーバーグループへのサーバーの追加およびサーバーグループからのサーバーの削除を可能にします。	ILB は、サーバーグループへのサーバーの追加またはサーバーグループからのサーバーの削除を動的に実行します。	NAT モード
セッション持続性 (固定性) を構成できるようにします。	ILB では、接続またはパケットがアプリケーションによってクライアントから同じバックエンドサーバーに送信されるように、セッション持続性を構成できます。ILB では、`-p` オプションを使用し、`ilbadm create-rule` コマンドで `pmask` オプションを指定することによって、仮想サービスのセッション持続性 (つまり、発信元アドレスの持続性) を構成でき	DSR および NAT の両方のモード

機能	説明	動作のモード
	ます。詳細は、70 ページの「ILB 規則の作成」を参照してください。	
接続排出を実行できるようにします。	ILB は、無効にされているサーバーに対して新しい接続が送信されることを回避します。この機能は、アクティブな接続またはセッションを中断せずにサーバーをシャットダウンする場合に役立ちます。サーバーへの既存の接続は動作を継続します。そのサーバーへのすべての接続が終了したあとに、サーバーは保守のためにシャットダウンできます。サーバーがリクエストを処理するための準備が整ったら、サーバーを有効にすることで、ロードバランサは新しい接続をサーバーに転送できるようになります。	NAT モード
トランスミッションコントロールプロトコル (TCP) およびユーザーデータグラムプロトコル (UDP) ポートの負荷分散を可能にします。	ILB は、各ポートに明示的な規則を設定しなくても、特定の IP アドレス上のすべてのポートを異なる一連のサーバーで負荷分散できます。	DSR および NAT の両方のモード
同じサーバーグループ内の仮想サービスに独立したポートを指定できるようにします。	ILB では、同じサーバーグループのさまざまなサーバーに異なる着信先ポートを指定できます。	NAT モード
単純なポート範囲で負荷分散できるようにします。	ILB では、VIP のポートの範囲を特定のサーバーグループに負荷分散します。便宜上、同一 VIP 上の異なるポート範囲を異なる一連のバックエンドサーバーで負荷分散することによって、IP アドレスを節約できます。また、NAT モードでセッション持続性が有効なときは、ILB は同一のクライアント IP アドレスから、範囲内のさまざまなポートへのリクエストを、同一のバックエンドサーバーに送信します。	DSR および NAT の両方のモード
ポート範囲を移動および縮小できるようにします。	ポート範囲の移動および縮小は、負荷分散規則のサーバーのポート範囲によって異なります。サーバーのポート範囲が VIP ポート範囲と異なる場合、ポート移動が自動的に実装されます。サーバーのポート範囲が単一ポートの場合、ポート収縮が実装されます。	NAT モード

ILB のコンポーネント、動作モード、アルゴリズム、および ILB の仕組みについては、第5章「統合ロードバランサの概要」を参照してください。ILB の構成および管理の詳細は、第6章「統合ロードバランサの構成と管理」および第7章「ILB の高可用性の構成」を参照してください。

VRRP ルーターおよびロードバランサを使用する理由

ローカルエリアネットワーク (LAN) などのネットワークを設定する場合、可用性の高いサービスを提供することは非常に重要です。高可用性は、障害の発生時にビジネス継続性を確保する

ために、冗長システムが引き継ぐ状態です。高可用性は、過剰なワークロードが冗長システムにオフロードされる場合にも関係します。高可用性は、計画または予想外の停止時間、負荷分散、障害回復などの状況で重要になります。

ネットワークドメイン内では、高可用性は、リンク、IP、ルーターなどのさまざまなレベルで実装できます。ロードバランサおよびルーターは、高可用性サービスを提供するために重要な役割を果たします。Oracle Solaris では、VRRP ルーターおよび ILB は、高可用性を提供するネットワークレベルのファイルオーバーと負荷共有メカニズムになります。

VRRP ルーターの操作および構成の詳細は、第3章「仮想ルーター冗長プロトコルの使用」を参照してください。ILB の操作および構成の詳細は、第5章「統合ロードバランサの概要」および第6章「統合ロードバランサの構成と管理」を参照してください。

♦ ♦ ♦ 第 2 章 2

ルーターとしてのシステムの構成

ルーターは、複数のネットワーク間のインタフェースを提供します。ルーターの物理ネットワークインタフェースごとに、一意の名前と IP アドレスを割り当てる必要があります。各ルーターは、そのプライマリネットワークインタフェースに関連付けられているホスト名と IP アドレスに加えて、増設した各ネットワークインタフェースについて少なくとも 1 つずつ、一意な名前と IP アドレスを持つことになります。この章では、IPv4 ルーターまたは IPv6 ルーターとして Oracle Solaris システムを構成する方法について説明します。

この章の内容は、次のとおりです。

ネットワーク上で Oracle Solaris ホストのルーティングを構成する方法の詳細は、『Oracle Solaris 11.2 でのネットワークコンポーネントの構成と管理 』の「単一インタフェースシステムのルーティングの有効化」を参照してください。

ルーティングプロトコルの詳細は、10 ページの「ルーティングプロトコル」および『Oracle Solaris 11.2 でのネットワークコンポーネントの構成と管理 』の「IPv6 ルーティングについて」を参照してください。

IPv4 ルーターの構成

次の手順を使えば、物理インタフェースが 1 つだけのシステム (デフォルトではホスト) をルーターとして構成できます。『Oracle Solaris 11.2 での UUCP および PPP を使用したシリアルネットワークの管理 』の「ダイアルアップ PPP リンクの計画」で説明しているように、システムが PPP リンクの 1 つのエンドポイントとして機能する場合は、単一インタフェースのシステムをルーターとして構成できます。

▼ IPv4 ルーターの構成方法

次の手順では、ルーターのインストール後にルーターのインタフェースを構成していることを想定しています。

始める前に ルーターがネットワークに物理的にインストールされたら、ローカルファイルモードで動作するようにルーターを構成します。この構成により、ネットワーク構成サーバーがダウンしても、ルーターが確実にブートされるようになります。

1. **管理者になります。**

   詳細は、『Oracle Solaris 11.2 でのユーザーとプロセスのセキュリティー保護 』の「割り当てられている管理権利の使用」を参照してください。

2. **システム上の NIC に対して IP インタフェースを構成します。**

   ```
   # ipadm create-ip IP-interface
   ```

3. **次のいずれかのコマンドを選択して、有効な IP アドレスを持つ IP インタフェースを構成します。**

   - **静的アドレスを構成するには、次のコマンドを入力します。**

     ```
     # ipadm create-addr -a address [interface | addr-obj]
     ```

   - **非静的アドレスを構成するには、次のコマンドを入力します。**

     ```
     # ipadm create-addr -T address-type [interface | addr-obj]
     ```

   IP インタフェースの構成方法の詳細は、『Oracle Solaris 11.2 でのネットワークコンポーネントの構成と管理 』の第 3 章「Oracle Solaris での IP インタフェースとアドレスの構成および管理」を参照してください。

   システムがパケットをルーティングする必要があるネットワークの IP アドレスを、各 IP インタフェースに構成します。したがって、システムで `192.168.5.0` および `10.0.5.0` のネットワークに対応する場合、ネットワークごとに 1 つの NIC を構成する必要があります。

注意 - DHCP を使用するように IPv4 ルーターを構成する前に、DHCP の管理に精通しておく必要があります。

4. **各インタフェースのホスト名および IP アドレスを `/etc/inet/hosts` ファイルに追加します。**

たとえば、ルーターの 2 つのインタフェースに割り当てた名前を、それぞれ krakatoa および krakatoa-1 とします。/etc/inet/hosts ファイルのエントリは次のようになります。

```
192.168.5.1      krakatoa        #interface for network 192.168.5.0
10.0.5.1         krakatoa-1      #interface for network 10.0.5.0
```

5. **『Oracle Solaris 11.2 でのネットワークコンポーネントの構成と管理 』の「システムをローカルファイルモード用に構成する方法」の手順を実行し、このルーターをローカルファイルモードで実行するように構成します。**

6. **サブネットに分割されたネットワークにルーターが接続されている場合、ネットワーク番号とネットマスクを /etc/inet/netmasks ファイルに追加します。**

   たとえば、IPv4 アドレス表記法 (192.168.5.0 など) の場合は、次のように入力します。

   ```
   192.168.5.0    255.255.255.0
   ```

7. **ルーターで IPv4 パケット転送を使用可能にします。**

   ```
   # ipadm set-prop -p forwarding=on ipv4
   ```

8. **(オプション) ルーティングプロトコルを開始します。**

   次のいずれかのコマンドを使用します。

   ```
   # routeadm -e ipv4-routing -u
   ```

   ここでは、-e オプションで IPv4 ルーティングを有効にし、-u オプションで、実行中のシステムに現在の構成を適用します。

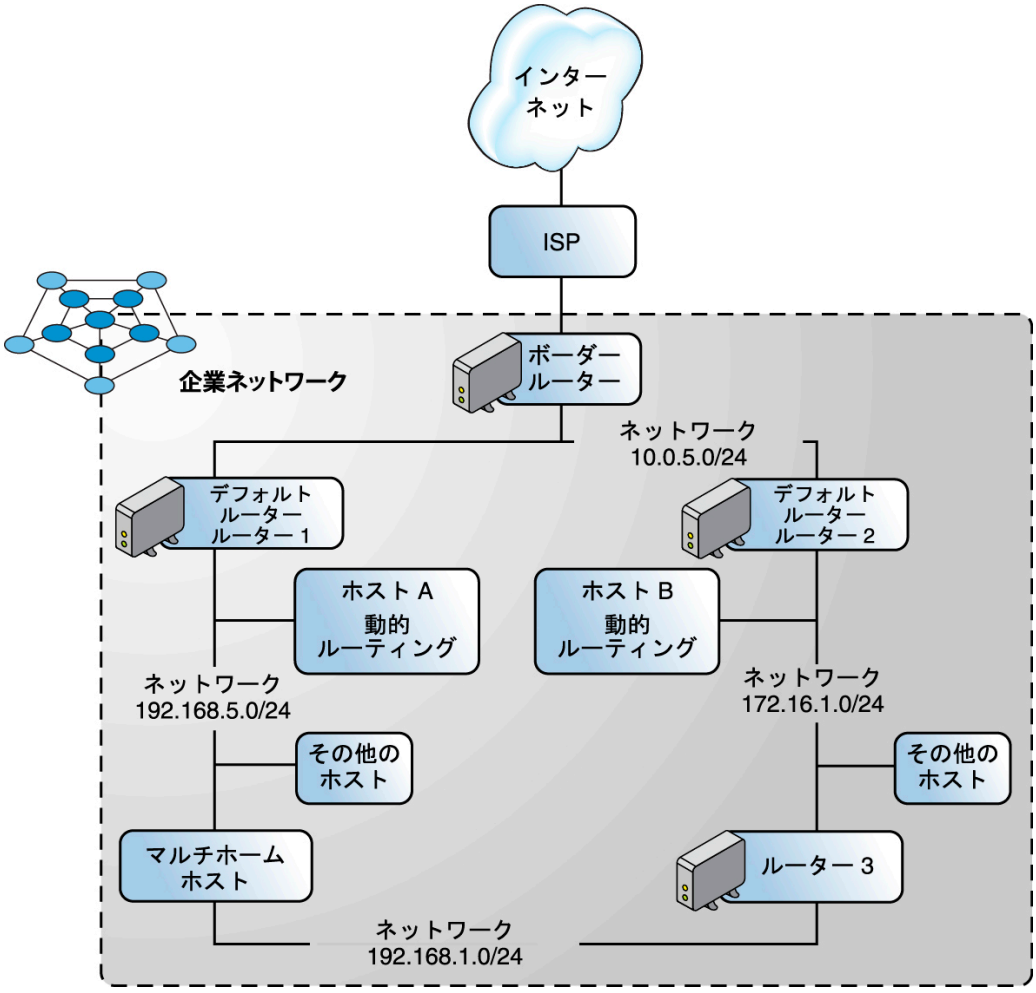
   ```
   # svcadm enable route:default
   ```

   ルーティングプロトコルを開始するときに、ルーティングデーモン /usr/sbin/in.routed は自動的にルーティングテーブルを更新し、このプロセスを*動的ルーティング*と呼びます。ルーティングのタイプの詳細は、『Oracle Solaris 11.2 でのネットワークコンポーネントの構成と管理 』の「ルーティングテーブルとルーティングの種類」を参照してください。routeadm コマンドの詳細は、routeadm(1M) のマニュアルページ、ipadm コマンドの詳細は、ipadm(1M) のマニュアルページを参照してください。

   in.routed デーモンに関連付けられているサービス管理機能 (SMF) の障害管理リソース識別子 (FMRI) は、svc:/network/routing/route. です。

例 2-1 ルーターとしてのシステムの構成

この例は、次の図に基づいています。



ルーター 2 には有線ネットワーク接続が 2 つあり、1 つはネットワーク `172.20.1.0`、もう 1 つはネットワーク `10.0.5.0` に接続されています。この例では、`172.20.1.0` ネットワークのルーター (ルーター 2) としてシステムを構成する方法を示しています。この例では、『Oracle Solaris 11.2 でのネットワークコンポーネントの構成と管理 』の「システムをローカルファイルモード用に構成する方法」に説明されているように、ルーター 2 がローカルファイルモードで動作するように構成されていることも想定しています。

1. システムのインタフェースのステータスを決定します。

```
# dladm show-link
LINK     CLASS     MTU     STATE   BRIDGE   OVER
net0     phys      1500    up      --       --
net1     phys      1500    up      --       --
net2     phys      1500    up      --       --
# ipadm show-addr
ADDROBJ           TYPE     STATE        ADDR
lo0/v4            static   ok           10.0.0.1/8
net0/v4           static   ok           172.20.1.10/24
```

2. net0 だけが IP アドレスで構成されています。ルーター 2 をデフォルトルーターにするには、net1 インタフェースを 10.0.5.0 ネットワークに物理的に接続します。

```
# ipadm create-ip net1
# ipadm create-addr -a 10.0.5.10/24 net1
# ipadm show-addr
ADDROBJ           TYPE     STATE        ADDR
lo0/v4            static   ok           192.168.0.1/8
net0/v4           static   ok           172.20.1.10/24
net1/v4           static   ok           10.0.5.10/24
```

3. 新たに構成したインタフェースとその接続先ネットワークの情報を使用して、次のネットワークデータベースを更新します。

```
# pfedit /etc/inet/hosts
192.168.0.1       localhost
172.20.1.10        router2        #interface for network 172.20.1
10.0.5.10          router2-out    #interface for network 10.0.5
# pfedit /etc/inet/netmasks
172.20.1.0    255.255.255.0
10.0.5.0      255.255.255.0
```

4. パケット転送と in.routed ルーティングデーモンを有効にします。

```
# ipadm set-prop -p forwarding=on ipv4
# svcadm enable route:default
```

これで、IPv4 パケット転送と RIP による動的ルーティングがルーター 2 で有効になりました。ただし、ネットワーク 172.20.1.0 のデフォルトルーターの構成を完了するには、次を実行する必要があります。

- 172.20.1.0 ネットワークの各ホストを変更して、ホストがルーティング情報をこの新しいデフォルトルーターから取得するようにします。詳細は、『Oracle Solaris 11.2 でのネットワークコンポーネントの構成と管理 』の「永続的 (静的) ルートの作成」を参照してください。
- ルーター 2 のルーティングテーブルで、ボーダールーターへの静的ルートを定義します。詳細は、『Oracle Solaris 11.2 でのネットワークコンポーネントの構成と管理 』の「ルーティングテーブルとルーティングの種類」を参照してください。ipadm コマンドの詳細は、ipadm(1M) のマニュアルページを参照してください。

IPv6 ルーターの構成

このセクションでは、IPv6 ルーターの構成方法について説明します。

in.ripngd デーモン、IPv6 ルーティング用

in.ripngd デーモンは、IPv6 ルーターの次世代ルーティング情報プロトコル (RIPng) を実装します。RIPng は IPv6 における RIP 相当機能を定義します。routeadm コマンドで IPv6 ルーターを構成し、IPv6 ルーティングを有効にした場合、in.ripngd デーモンはそのルーターに RIPng を実装します。RIPng のサポートされるオプションの詳細は、in.ripngd(1M) を参照してください。

ルーター広告、接頭辞およびメッセージ

マルチキャスト対応リンクとポイントツーポイントリンクでは、各ルーターは定期的にルーター広告パケットをマルチキャストグループに送信して、ルーターが利用できることを知らせます。ホストはすべてのルーターからルーター広告を受け取り、デフォルトルーターのリストを作成します。ルーターは頻繁にルーター広告を生成するので、ホストは数分でルーターが利用できることを知ることができます。ただし、通知がないからといってルーターエラーであると判断できるほどの頻度ではありません。エラー検出には、近傍到達不能性を判別する別の検出アルゴリズムを利用します。

ルーター広告には、ホストがルーターと同じリンク上にある (つまり、オンリンクである) かどうかを判断するときに使用するサブネット接頭辞のリストが含まれます。この接頭辞リストは、自動ア

ドレス構成にも使用されます。接頭辞に付属するフラグは特定の接頭辞の使用目的を表します。ホストは通知されたオンリンク接頭辞を使用して、パケットの宛先がオンリンクであるか、あるいはルーターを越えているかを判断するためのリストを作成および管理します。通知されたオンリンク接頭辞になくても宛先がオンリンクの場合があります。この場合、ルーターはリダイレクトを送ることができます。リダイレクトは送信側に、宛先が近傍であることを知らせます。

ルーター広告と接頭辞別のフラグを使用すると、ルーターはステートレスアドレス自動構成を実行する方法をホストに伝えることができます。

ルーター広告メッセージには、ホストが発信するパケットに使用するインターネットパラメータ (ホップの制限など) も含めることができます。また、オプションでリンク MTU などのリンクパラメータも含めることができます。この機能により、重要なパラメータを集中管理できます。パラメータは、ルーターに設定され、関連付けられたすべてのホストに自動的に伝達されます。

アドレス解決を行うために、ノードは、宛先ノードがリンク層アドレスを戻すように要求する近傍要請をマルチキャストグループに送信します。マルチキャストされた近傍要請メッセージは、宛先アドレスの要請先ノードのマルチキャストアドレスに送信されます。宛先は、そのリンク層アドレスをユニキャスト近傍通知メッセージで戻します。発信元と宛先の両方に対して 1 つの要求応答パケットペアで互いのリンク層アドレスを処理できます。発信元は、近傍要請に発信元のリンク層アドレスを組み込みます。

▼ IPv6 対応のルーターを構成する方法

次の手順では、システムがすでに IPv6 用に構成されているものとします。手順の詳細は、『Oracle Solaris 11.2 でのネットワークコンポーネントの構成と管理 』の第 3 章「Oracle Solaris での IP インタフェースとアドレスの構成および管理」を参照してください。

1. **管理者になります。**

   詳細は、『Oracle Solaris 11.2 でのユーザーとプロセスのセキュリティー保護 』の「割り当てられている管理権利の使用」を参照してください。

2. **ルーターのすべてのインタフェース上で IPv6 パケット転送を構成します。**

   ```
   # ipadm set-prop -p forwarding=on ipv6
   ```

3. **ルーティングデーモンを起動します。**

   `in.ripngd` デーモンは IPv6 ルーティングを処理します。次のいずれかのコマンドを使用して、IPv6 ルーティングを有効にします。

- routeadm コマンドを次のように使用します。

  ```
  # routeadm -e ipv6-routing -u
  ```

  ここでは、-e オプションで IPv4 ルーティングを有効にし、-u オプションで、実行中のシステムに現在の構成を適用します。
- 適切な SMF コマンドを次のように使用します。

  ```
  # svcadm enable ripng:default
  ```

routeadm コマンドの詳細は、routeadm(1M) のマニュアルページを参照してください。

4. **/etc/inet/ndpd.conf ファイルを作成します。**

   /etc/inet/ndpd.conf ファイルには、ルーターが通知するサイト接頭辞などの構成情報を指定します。このファイルを in.ndpd デーモンが読み取って、IPv6 近傍検察プロトコルを実装します。

   変数と指定できる値のリストについては、ndpd.conf(4) のマニュアルページを参照してください。

5. **次のテキストを /etc/inet/ndpd.conf ファイルに入力します。**

   ```
   ifdefault AdvSendAdvertisements true
   prefixdefault AdvOnLinkFlag on AdvAutonomousFlag on
   ```

   このテキストは、ルーターの IPv6 用に構成されたすべてのインタフェース経由で、ルーター広告を送信することを in.ndpd デーモンに指示します。

6. **ルーターのさまざまなインタフェースにサイト接頭辞を構成するには、追加のテキストを /etc/inet/ndpd.conf ファイルに追加します。**

   テキストは次の形式で追加するようにしてください。

   prefix *global-routing-prefix*:*subnet ID*/64 *interface*

   次の例の /etc/inet/ndpd.conf ファイルは、サイト接頭辞 2001:0db8:3c4d::/48 をインタフェース net0 および net1 経由で通知するようにルーターを構成します。

   ```
   ifdefault AdvSendAdvertisements true
   prefixdefault AdvOnLinkFlag on AdvAutonomousFlag on

   if net0 AdvSendAdvertisements 1
   prefix 2001:0db8:3c4d:15::0/64 net0
   ```

```
if net1 AdvSendAdvertisements 1
prefix 2001:0db8:3c4d:16::0/64 net1
```

7. **システムをリブートします。**

   IPv6 ルーターは、`ndpd.conf` ファイルにあるサイト接頭辞をローカルリンクに通知し始めます。

8. **IPv6 用に構成されたインタフェースを表示します。**

```
# ipadm show-addr
ADDROBJ       TYPE       STATE   ADDR
lo0/v4        static     ok      192.68.0.1/8
net0/v4       static     ok      172.16.15.232/24
net1/v4       static     ok      172.16.16.220/24
net0/v6       addrconf   ok      fe80::203:baff:fe11:b115/10
lo0/v6        static     ok      ::1/128
net0/v6a      static     ok      2001:db8:3c4d:15:203:baff:fe11:b115/64
net1/v6       addrconf   ok      fe80::203:baff:fe11:b116/10
net1/v6a      static     ok      2001:db8:3c4d:16:203:baff:fe11:b116/64
```

   この出力では、IPv6 用に構成されている各インタフェースは、この時点で 2 つのアドレスを持っています。*interface*`/v6` のようなアドレスオブジェクト名を含むエントリには、そのインタフェースのリンクローカルアドレスが表示されています。*interface*`/v6add` のようなアドレスオブジェクト名を含むエントリには、グローバル IPv6 アドレスが表示されています。このアドレスには、インタフェース ID に加えて、`/etc/ndpd.conf` ファイルに構成されているサイト接頭辞が含まれます。`v6a` という指定はランダムに定義された文字列です。`net0/mystring` や `net0/ipv6addr` のように、*interface* が IPv6 アドレスの作成先となるインタフェースを表しているかぎり、アドレスオブジェクト名の 2 番目の部分としてほかの文字列を定義できます。

参照

- IPv6 ネットワークトポロジで識別されたルーターからのトンネルを構成する方法を見つけるには、『Oracle Solaris 11.2 での TCP/IP ネットワーク、IPMP、および IP トンネルの管理 』の「IP トンネルの管理」を参照してください。
- ネットワーク上のスイッチやハブを構成する方法については、スイッチまたはハブに付属するドキュメントを参照してください。
- サーバーの IPv6 サポートを改善する方法を見つけるには、『Oracle Solaris 11.2 でのネットワークコンポーネントの構成と管理 』の「サーバー上での IPv6 が有効なインタフェースの構成」を参照してください。

♦ ♦ ♦ 第 3 章

3 仮想ルーター冗長プロトコルの使用

ネットワークの信頼性を高める方法の 1 つは、ネットワーク内の重要なコンポーネントのバックアップを提供することです。Oracle Solaris では、高可用性を実現するために、仮想ルーター冗長プロトコル (VRRP) の使用を構成および管理する管理ルーツが用意されています。VRRP は、RFC 5798 (http://www.rfc-editor.org/rfc/rfc5798.txt) に指定されているインターネット標準プロトコルです。

Oracle Solaris 11.2 では、独自のレイヤー 3 VRRP により、IPMP および InfiniBand インタフェース経由の VRRP ルーターの作成をサポートし、ゾーンの VRRP の既存のサポートを拡張します。

注記 - この章を通して、レイヤー 2 VRRP (L2 VRRP) の用語のすべての参照は、特にインターネット標準 VRRP を指し、レイヤー 3 VRRP (L3 VRRP) の用語のすべての参照は、独自の Oracle Solaris レイヤー 3 VRRP を指します。

この章では、Oracle Solaris のレイヤー 2 VRRP および独自のレイヤー 3 VRRP の概要について説明します。

この章の内容は、次のとおりです。

VRRP について

VRRP は、ルーターやロードバランサに使用される IP アドレスなど、IP アドレスの高可用性を提供します。サービスが負荷分散などのルーティング以外の機能を提供する場合でも、VRRP を

使用するサービスは VRRP ルーターとも呼ばれます。高可用性を実現するために、ロードバランサで VRRP がどのように使用されるかについては、第7章「ILB の高可用性の構成」を参照してください。

VRRP ルーターは、VRRP を実行しているルーターです。VRRP は各 VRRP ルーター上で実行され、そのルーターの状態を管理します。1 つのホストは、VRRP が構成された複数のルーターを持つことができ、各 VRRP ルーターは異なる仮想ルーターに属します。VRRP を使用してローカルエリアネットワーク (LAN) に仮想ルーターを導入すると、ルーターの障害回復を実現できます。

VRRP の動作

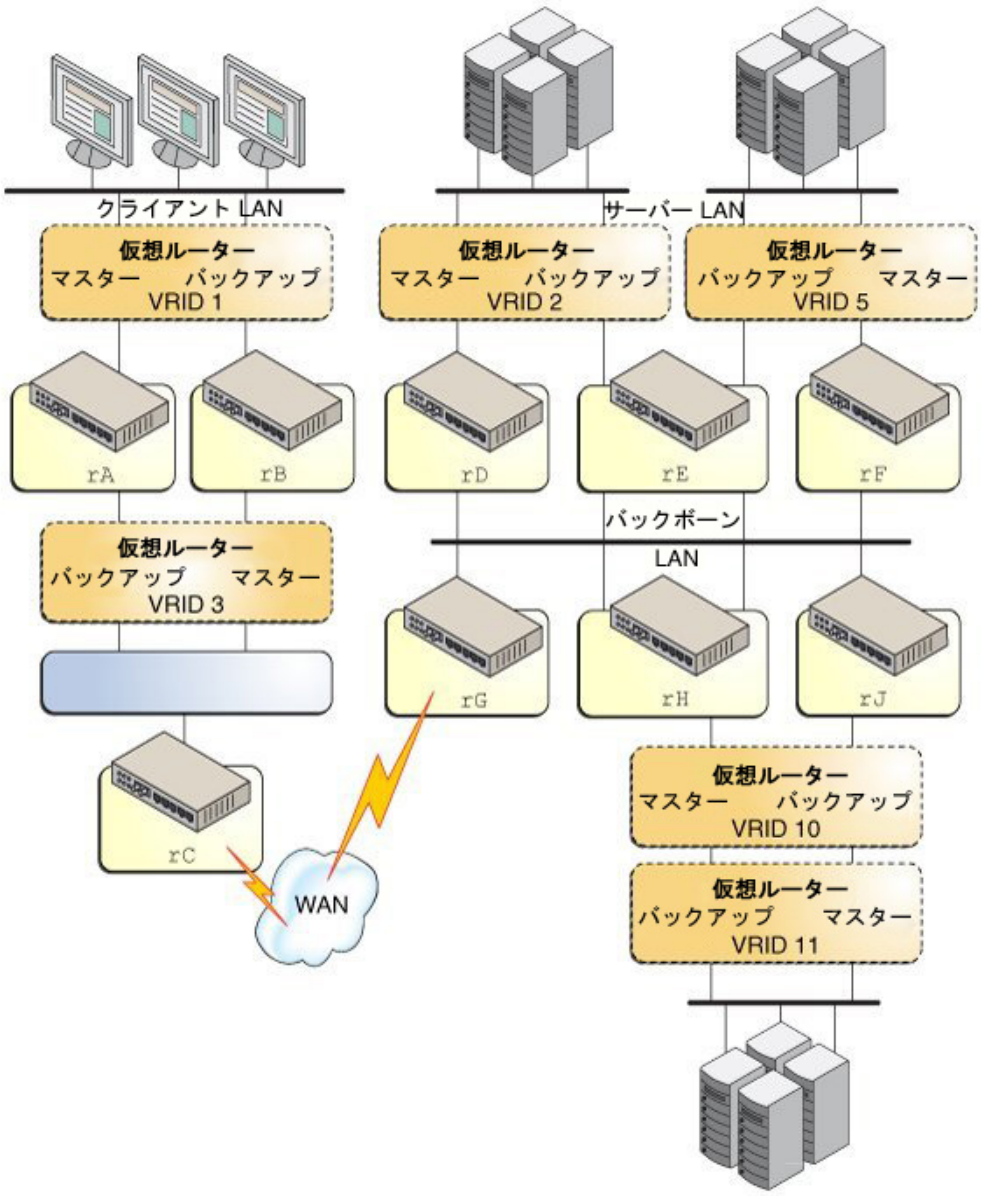
次の VRRP ルーターの用語に注意してください。

- ルーター名 – システム全体で一意の識別子。
- 仮想ルーター ID (VRID)– 特定のネットワークセグメント上の仮想ルーターを識別するために使用される一意の番号。VRID は LAN 内の仮想ルーターを識別します。
- プライマリ IP アドレス – VRRP 通知の発信元 IP アドレス。
- 仮想 IP アドレス (VRIP)– VRID に関連付けられる IP アドレス。ほかのホストはそこからネットワークサービスを取得できます。VRIP は、VRID に属する VRRP インスタンスによって管理されます。
- マスタールーター – 特定の時間に仮想ルーターのルーティング機能を実行する VRRP インスタンス。ある特定の VRID である瞬間にアクティブになっている*マスタールーター*は、1 つだけです。マスタールーターは、仮想ルーターに関連付けられた 1 つまたは複数の IPv4 または IPv6 アドレスを制御します。仮想ルーターは、マスタールーターの IP アドレスに送信されたパケットを転送します。
- バックアップルーター – アクティブだが、*バックアップルーター*と呼ばれるマスター状態にはない VRID の VRRP インスタンス。VRID には、任意の数のバックアップルーターが存在できます。現在のマスタールーターで障害が発生した場合、バックアップルーターはそのマスタールーターの役割になれます。
- VRRP パラメータ – 優先順位、通知間隔、プリエンプションモード、および受け入れモードを含みます。
- VRRP の状態情報と統計。

次の VRRP 負荷共有構成図は、単一のルーターインタフェース上に複数の VRID が存在できることを示しています。図で使用される VRRP コンポーネントについては、付随するテキスト

で説明します。この VRRP 負荷共有構成は、単一のルーターインタフェース上に複数の VRID が存在できることを示しています。

図 3-1 LAN での VRRP の負荷共有構成



- ルーター rA は、仮想ルーター VRID 1 のマスタールーターであり、VRID 3 のバックアップルーターです。ルーター rA は、VRID 1 の仮想 IP (VIP) アドレスにアドレス指定されたパケットのルーティングを処理し、VRID 3 のルーティング役割になる準備が整っています。
- ルーター rB は、仮想ルーター VRID 3 のマスタールーターであり、VRID 1 のバックアップルーターです。ルーター rB は、VRID 3 の VIP にアドレス指定されたパケットのルーティングを処理し、VRID 1 のルーティングの役割を担う準備が整っています。
- ルーター rC は、VRRP 機能を持っていませんが、VRID 3 の VIP を使用してクライアント LAN のサブネットに到達します。
- ルーター rD は VRID 2 のマスタールーターです。ルーター rF は VRID 5 のマスタールーターです。ルーター rE は、これらの VRID の両方に対するバックアップルーターです。rD または rF で障害が発生すると、rE がその VRID のマスタールーターになります。rD と rF の両方で同時に障害が発生する可能性があります。VRRP ルーターは、1 つ以上の VRID のマスタールーターになることができます。
- ルーター rG は、バックボーン LAN の広域ネットワーク (WAN) ゲートウェイです。バックボーンに接続されているルーターはすべて、OSPF などの動的ルーティングプロトコルを使用して WAN 上のルーターとルーティング情報を共有しています。VRRP はこの点に関与しませんが、ルーター rC は、クライアント LAN のサブネットへのパスが VRID 3 の VIP 経由であることを通知します。
- ルーター rH は、VRID 10 のマスタールーターであり、VRID 11 のバックアップルーターです。同様に、ルーター rJ は VRID 11 のマスタールーターであり、VRID 10 のバックアップルーターです。

レイヤー 3 VRRP 機能について

Oracle Solaris の独自のレイヤー 3 仮想ルーター冗長プロトコル (L3 VRRP) 機能では、VRRP ルーターの一意の VRRP 仮想 MAC アドレスの構成が不要なため、IPMP および InfiniBand インタフェース経由、またゾーン内での VRRP のサポートが向上します。L3 VRRP プロトコルは、標準の VRRP 仕様に準拠していません。L3 VRRP 実装では、同じ仮想ルーターの VRRP ルーター間で一意の仮想 MAC アドレスを使用する代わりに、Gratuitous アドレス解決プロトコル (ARP) メッセージおよび近傍検索プロトコル (NDP) メッセージを使用して、現在のマスター VRRP ルーターの仮想 IP アドレスと MAC アドレス間のマッピングをリフレッシュします。

レイヤー 3 VRRP では、IPMP および InfiniBand インタフェース経由の VRRP のサポート、およびゾーン内のサポートの強化というメリットが得られ、VRRP VNIC の作成も不要です。

レイヤー 2 およびレイヤー 3 VRRP の比較

次の表は、レイヤー 2 とレイヤー 3 VRRP を比較したものです。

表 3-1 レイヤー 2 とレイヤー 3 VRRP の比較

機能	レイヤー 2 VRRP	レイヤー 3 VRRP
VRRP VNIC の作成	VRRP VNIC を作成する必要があります。	VRRP VNIC によって提供される仮想 VRRP MAC アドレスは必要ないため、VRRP VNIC の作成は不要です。
IPMP のサポート	サポートされていません。	サポートされています。レイヤー 3 VRRP ルーターが IPMP グループインタフェース経由で作成されると、マスタールーターの各仮想 IP アドレスは、既存の IPMP ポリシーに従い、基礎となるアクティブな IPMP インタフェースの MAC アドレスに関連付けられます。IPMP グループでフェイルオーバーが発生すると、L2 または L3 マッピングが Gratuitous ARP または NDP メッセージによって通知されます。
ゾーンのサポート	同じ仮想ルーターに属する複数の VRRP ルーターを異なるゾーンで実行すると問題が発生します。同じ VRRP 仮想 MAC アドレスを共有する 2 つ以上の VRRP ルーターを備えたシステムでは、組み込みの仮想スイッチによって、VRRP ルーターへの通常の VRRP 通知パケットのフローが中断されます。詳細は、33 ページの「レイヤー 2 およびレイヤー 3 VRRP の制限事項」を参照してください。	サポートされています。
InfiniBand サポート	サポートされていません。	サポートされています。
一意の仮想ルーター MAC アドレス	一意の仮想ルーター MAC アドレスが必要です。仮想 IP アドレスは、常に同じ仮想 MAC アドレスに解決されます。	不要。VRRP ルーターが作成される MAC アドレスを使用します。MAC アドレスは、同じ仮想ルーター内のすべての VRRP ルーター間で異なります。同じ MAC アドレスは、この L3 VRRP ルーターで保護される仮想 IP アドレスに関連付けられます。

機能	レイヤー 2 VRRP	レイヤー 3 VRRP
VRRP 仮想 IP アドレスの構成	構成が必要。	構成が必要。
インターネット制御メッセージプロトコル (ICMP) リダイレクト	L2 VRRP がルーターのグループ間で実行されている場合に使用される場合があります。L2 VRRP ルーターで ICMP リダイレクトを使用する必要がある場合は、リダイレクトする必要のあるパケットの宛先 MAC アドレス (VRRP 仮想 MAC アドレス) がチェックされます。宛先 MAC アドレスを使用することで、L2 VRRP ルーターは、パケットが最初に送信された仮想ルーターを判断します。これにより、L2 VRRP ルーターは、発信元アドレスを選択し、ICMP リダイレクトメッセージを発信元に送信できます。	ICMP リダイレクトを無効にする必要があります。複数の VRRP ルーターが同じインタフェースで作成されると、同じ MAC アドレスが共有されます。このため、L3 VRRP は宛先 MAC アドレスを判断できません。
マスタールーターの選択	マスタールーターの選択はホストに透過的です。マスタールーターが変更されると、ホストとルーターの間に存在するスイッチは、MAC 学習機能を使用して、トラフィックを送信する新しいポートを識別します。	マスタールーターの選択により、仮想 IP アドレスのレイヤー 2 マッピングが変更され、新しいマッピングは、Gratuitous ARP または NDP メッセージによって通知する必要があります。
フェイルオーバー時間	標準。	マスタールーターの選択が変更される場合、Gratuitous ARP または NDP メッセージの追加要件により、長くなる場合があります。

レイヤー 2 およびレイヤー 3 VRRP の制限事項

レイヤー 2 およびレイヤー 3 の VRRP には、共通の制限事項があり、レイヤー 2 およびレイヤー 3 VRRP 仮想 IP アドレスを静的に構成する必要があります。IP アドレスの既存の 2 つの自動構成ツール (`in.ndpd` (IPv6 自動構成用) および `dhcpagent` (動的ホスト構成プロトコル (DHCP) 構成用)) を使用して、VRRP 仮想 IP アドレスを自動構成することはできません。また、レイヤー 2 およびレイヤー 3 VRRP には、特定の制限事項があります。

レイヤー 2 VRRP 機能には、次の制限事項があります。

- **排他的 IP ゾーンのサポート**

  VRRP ルーターが排他的 IP ゾーンに作成されると、VRRP サービス `svc:/network/vrrp/default` が自動的に有効になります。その VRRP サービスが、その特定のゾーンの VRRP ルーターを管理します。ただし、排他的 IP ゾーンのサポートは次のように制限されます。

- 仮想ネットワークインタフェースカード (VNIC) は非大域ゾーンに作成できないため、VRRP VNIC を最初に大域ゾーンに作成する必要があります。次に、VRRP ルーターが存在する非大域ゾーンに VNIC を割り当てる必要があります。これにより、`vrrpadm` コマンドを使用して、VRRP ルーターを非大域ゾーンに作成できます。
- 単一の Oracle Solaris システムでは、2 つの VRRP ルーターを異なるゾーンに作成して、同じ仮想ルーターに参加させることはできません。Oracle Solaris では、同じメディアアクセス制御 (MAC) アドレスの 2 つの VNIC を作成することはできません。

- **その他のネットワーク機能との相互運用**
  - L2 VRRP サービスは、IP ネットワークマルチパス (IPMP) インタフェース上で動作できません。VRRP は特定の VRRP MAC アドレスを必要としますが、IPMP は完全に IP レイヤーで動作します。IPMP については、『Oracle Solaris 11.2 での TCP/IP ネットワーク、IPMP、および IP トンネルの管理 』の第 2 章「IPMP の管理について」を参照してください。

    VRRP は、トランクまたは DLMP アグリゲーションモードのリンクアグリゲーションでは使用できません。アグリゲーションの詳細は、『Oracle Solaris 11.2 でのネットワークデータリンクの管理 』の第 2 章「リンクアグリゲーションを使用した高可用性の構成」を参照してください。
  - L2 VRRP サービスは、IP over Infiniband (IPMP) インタフェースでは動作できません。
- **Ethernet over InfiniBand のサポート**

  L2 VRRP では、Ethernet over InfiniBand (EoIB) インタフェースはサポートしていません。すべての L2 VRRP ルーターは一意の仮想 MAC アドレスに関連付けられているため、同じ仮想ルーターで参加している VRRP ルーターは、(EoIB インタフェースではサポートされない) 同じ仮想 MAC アドレスを同時に使用する必要があります。L3 VRRP では、同じ仮想ルーターに存在するすべての VRRP ルーター間で異なる MAC アドレスを使用するため、この制限事項が解決されます。

レイヤー 3 VRRP 機能には、次の制限事項があります。

- Gratuitous ARP または NDP メッセージにより、マスタールーターの選択中のフェイルオーバー時間が長くなる場合があります。

  L3 VRRP では、マスタールーターの選択が変更されると、Gratuitous ARP または NDP メッセージを使用して、新しい L2 または L3 マッピングを通知します。Gratuitous ARP または NDP メッセージのこの追加要件により、フェイルオーバー時間が長くなる場合があります。通知されたすべての Gratuitous ARP または NDP メッセージが失われる場合は、リフレッシュされた ARP または NDP エントリをホストが受信するのに時間がかかる場合が

あります。したがって、新しいマスタールーターへのパケットの送信が遅延する場合があります。

- 複数のルーターで同じ宛先 MAC アドレスが共有されているため、ICMP リダイレクトの使用時に宛先 MAC アドレスを判断できません。

  対称でないネットワークトポロジのルーターのグループ間で VRRP を使用している場合は、ICMP リダイレクトを使用できます。ICMPv4 リダイレクトまたは ICMPv6 リダイレクトの IPv4 または IPv6 ソースアドレスは、次のホップルーティングの決定時に、エンドホストによって使用されるアドレスである必要があります。

  L3 VRRP ルーターで ICMP リダイレクトを使用する必要がある場合は、リダイレクトする必要のあるパケットの宛先 MAC アドレス (VRRP 仮想 MAC アドレス) が L3 VRRP ルーターによってチェックされます。同じインタフェース経由で作成される複数のルーターによって、同じ宛先 MAC アドレスが共有されるため、L3 VRRP ルーターでは、宛先 MAC アドレスを判断することはできません。したがって、L3 VRRP ルーターを使用する場合は、ICMP リダイレクトを無効にすると有効な場合があります。次のように、`send_redirects` パブリック IPv4 および IPv6 プロトコルのプロパティーを使用して、ICMP リダイレクトを無効にできます。

  ```
  # ipadm set-prop -m ipv4 -p send_redirects=off
  ```

- VRRP 仮想 IP アドレスは、`in.ndpd` または DHCP のいずれでも自動的に構成できません。

♦ ♦ ♦ 第 4 章

4 仮想ルーター冗長プロトコルの構成および管理

この章では、レイヤー 2 およびレイヤー 3 VRRP を構成するためのタスクについて説明します。この章の内容は、次のとおりです。

VRRP 構成の計画

レイヤー 2 またはレイヤー 3 VRRP の構成を計画する手順は次のとおりです。

1. L2 VRRP または L3 VRRP ルーターを構成するかどうかを決定する。
2. (L2 VRRP ルーターの場合のみ) VRRP VNIC を作成する。詳細は、39 ページの「レイヤー 2 VRRP の VRRP VNIC の作成」を参照してください。

   L2 VRRP ルーターの作成中に、`vrrpadm` コマンドの `-f` オプションを使用して、VRRP VNIC を自動的に作成できます。
3. VRRP ルーターを作成する。詳細は、40 ページの「VRRP ルーターの作成」を参照してください。
4. VRRP ルーターの仮想 IP アドレスを構成する。詳細は、42 ページの「レイヤー 2 および 3 VRRP ルーターの仮想 IP アドレスの構成」を参照してください。

   `vrrpadm` コマンドの `-a` オプションを使用して、仮想 IP アドレスを構成できます。詳細は、40 ページの「VRRP ルーターの作成」を参照してください。

VRRP のインストール

VRRP をシステムで使用するには、VRRP をインストールする必要があります。

▼ VRRP のインストール方法

1. **管理者になります。**

   詳細は、『Oracle Solaris 11.2 でのユーザーとプロセスのセキュリティー保護 』の「割り当てられている管理権利の使用」を参照してください。

2. **VRRP パッケージがインストールされているかどうかを確認します。**

   ```
   # pkg info vrrp
   ```

3. **VRRP パッケージがインストールされていない場合、インストールします。**

   ```
   # pkg install vrrp
   ```

VRRP の構成

`vrrpadm` コマンドを使用して、VRRP ルーターを構成できます。`vrrpadm` コマンドのすべてのサブコマンドの結果は、`vrrpadm show-router` コマンドを除き永続します。たとえば、`vrrpadm create-router` コマンドによって作成される VRRP ルーターは、リブートしても永続します。詳細は、`vrrpadm`(1M) のマニュアルページを参照してください。

VRRP ルーターを構成するには、ネットワーク管理プロファイルの一部になる `solaris.network.vrrp` 承認が必要です。

注記 - `vrrpadm show-router` コマンドによって開始される読み取り専用の操作では、`solaris.network.vrrp` 承認は不要です。

注意 - Oracle Solaris バンドルの IP フィルタで VRRP を使用する場合は、`ipfstat -io` コマンドを使用して、受信または送信 IP トラフィックが標準の VRRP マルチキャストアドレスの `224.0.0.18/32` で許可されているかどうかをチェックする必要があります。トラフィックが許可されていない場合、マスターおよびバックアップの両方の VRRP ルーターは `MASTER` 状態になります。したがって、VRRP ルーターごとに、対応する規則を IP フィルタ構成に追加する必要があります。詳細は、『Oracle Solaris 11.2 でのネットワーク管理のトラブルシューティング 』の「VRRP と Oracle Solaris バンドル版 IP フィルタに関する問題のトラブルシューティング」を参照してください。

レイヤー 2 VRRP の VRRP VNIC の作成

VNIC は、システムの物理ネットワークアダプタの上部に構成される仮想ネットワークインタフェースで、ネットワークの仮想化に不可欠なコンポーネントです。1 つの物理インタフェースが複数の VNIC を持つことができます。VNIC の詳細は、『Oracle Solaris 11.2 での仮想ネットワークとネットワークリソースの管理 』を参照してください。

各レイヤー 2 VRRP ルーターには、専用の VRRP VNIC が必要です。次のコマンド構文を使用します。

```
# dladm create-vnic [-t] [-R root-dir] -l link [-m vrrp -V VRID -A \
{inet | inet6}] [-v VLAN-ID] [-p prop=value[,...]] VNIC
```

このコマンドは、VRRP 仕様で定義されている仮想ルーター MAC アドレスを持つ VNIC を作成します。VNIC アドレスを使用して、`vrrp` を入力し、VRID およびアドレスファミリを指定します。アドレスファミリは、`inet` (IPv4 アドレス) または `inet6` (IPv6 アドレス) のいずれかになります。例:

```
# dladm create-vnic -m vrrp -V 21 -A inet6 -l net0 vnic0
```

詳細は、`dladm`(1M) のマニュアルページを参照してください。

注記 - `vrrpadm` コマンドで `-f` オプションを指定して、VRRP VNIC を作成することもできます。詳細は、40 ページの「VRRP ルーターの作成」を参照してください。

VRRP ルーターの作成

`vrrpadm create-router` コマンドは、指定された VRID とアドレスファミリ、およびその他の指定されたパラメータを持つレイヤー 2 またはレイヤー 3 VRRP ルーターを作成します。詳細は、`vrrpadm`(1M) のマニュアルページを参照してください。

VRRP ルーターを作成するには、次の構文を使用します。

```
# vrrpadm create-router [-T {l2 | l3}] [-f] -V VRID -I ifname \
-A [inet | inet6] [-a assoc-IPaddress] [-P primary-IPaddress] \
[-p priority] [-i adv-interval] [-o flags] router-name
```

`-T l2 | l3`
: ルーターのタイプを指定します。次の値のいずれかにタイプを設定できます。デフォルトは `l2` です。
  - `l2` – L2 タイプ VRRP ルーター
  - `l3`– L3 タイプ VRRP ルーター

`-f`
: (L2 VRRP のみ) L2 VRRP ルーターを持つ VRRP VNIC の作成を指定します。`-f` オプションを指定すると、`vrrpadm` コマンドは、指定された VRID およびアドレスファミリを持つ VRRP VNIC が存在するかどうかをチェックします。まだ存在しない場合にのみ、VRRP VNIC が作成されます。VRRP VNIC の名前は、命名規則 `vrrp-`*VRID_ifname*`_v`*4 | 6* を使用して生成されます。`-f` オプションは、レイヤー 3 VRRP ルーターの作成時に影響ありません。

`-V` *VRID*
: アドレスファミリに関連付けられている場合に、VLAN を定義する仮想ルーター識別子。

`-I` *ifname*
: VRRP ルーターが構成されるインタフェース。レイヤー 2 VRRP の場合、インタフェースは物理リンク、VLAN、またはアグリゲーションになります。レイヤー 3 VRRP の場合、インタフェースには、IPMP インタフェース、DHCP 管理インタフェース、および InfiniBand インタフェースを含めることもできます。このリンクにより、この VRRP ルーターが動作する LAN が決まります。

`-A [inet | inet6]`
: アドレスファミリ (`inet` (IPv4 アドレス) または `inet6` (IPv6 アドレス) のいずれか)。

`-a` *assoc-IPaddress*
: IP アドレスのコンマ区切りのリストを指定します。
  この場合、IP アドレスは次の書式のいずれかで指定できます。
  - *IP-address*[/*prefix-length*]

- *hostname*[/*prefix-length*]
- `linklocal`

  `linklocal` を指定すると、関連付けられた仮想ルーターの VRID に基づいて、IPv6 リンクローカル `vrrp` アドレスが構成されます。`linklocal` 形式は、IPv6 VRRP ルーターのみに適用されます。VNIC が自動的に作成および plumb されるように、`-a` オプションと `-f` オプションを組み合わせることができます。

`-P` *primary-IPaddress*
: VRRP 通知の送信に使用される VRRP プライマリ IP アドレスを指定します。

`-p` *priority*
: 指定された VRRP ルーターの、マスター選択で使用される優先度。デフォルト値は `255` です。優先度の値が最高のルーターがマスタールーターとして選択されます。

`-i` *adv-interval*
: 通知間隔 (ミリ秒)。デフォルト値は `1000` です。

`-o` *flags*
: VRRP ルーターのプリエンプションおよび受け入れモード。値は、`preempt` または `un_preempt`、`accept` または `no_accept` です。デフォルトでは、プリエンプションモードおよび受け入れモードは、それぞれ `preempt` および `accept` に設定されます。

router-name
: *router-name* は、この VRRP ルーターの一意の識別子です。ルーター名に使用できる文字は、英数字 (`a-z`、`A-Z`、`0-9`) および下線 (_) です。ルーター名の最大長は 31 文字です。

例 4-1 レイヤー 2 VRRP ルーターの作成

次の例は、データリンク `net0` 経由のルーターの作成方法を示しています。

```
# dladm create-vnic -m vrrp -V 12 -A inet -l net0 vnic1
# vrrpadm create-router -V 12 -A inet -p 100 -I net0 l2router1
# vrrpadm show-router l2router1
NAME       VRID  TYPE   IFNAME AF   PRIO ADV_INTV MODE  STATE  VNIC
l2router1  12    L2     net0   IPv4 100  1000     e-pa- BACK   vnic1
```

L2 VRRP ルーター `l2router1` は、IPv4 アドレスファミリおよび VRID `12` を持つデータリンク `net0` 経由で作成されます。`vrrpadm show-router` コマンドについては、44 ページの「レイヤー 2 およびレイヤー 3 VRRP ルーター構成の表示」を参照してください。

例 4-2 レイヤー 3 VRRP ルーターの作成

次の例は、ipmp0 という名前の IPMP インタフェース経由で L3 VRRP ルーターを作成する方法を示しています。

```
# vrrpadm create-router -V 6 -I ipmp0 -A inet -T l3 l3router1
# vrrpadm show-router
NAME      VRID TYPE IFNAME AF   PRIO ADV_INTV MODE  STATE VNIC
l3router1 6    L3   ipmp0  IPv4 255  1000     eopa- INIT  --
```

L3 VRRP ルーター l3router1 は、IPv4 アドレスファミリおよび VRID 6 を持つ IPMP インタフェース ipmp0 経由で作成されます。vrrpadm show-router コマンドについては、44 ページの「レイヤー 2 およびレイヤー 3 VRRP ルーター構成の表示」を参照してください。

レイヤー 2 および 3 VRRP ルーターの仮想 IP アドレスの構成

L2 VRRP ルーターの IP アドレスを構成するには、関連付けられている VRRP VNIC 経由で、タイプ vrrp の仮想 IP アドレスを構成する必要があります。

L3 VRRP ルーターの仮想 IP アドレスを構成するには、L3 VRRP ルーターが構成されている同じ IP インタフェースで、タイプ vrrp の IP アドレスを使用する必要があります。

注記 - IPv6 アドレスを構成するには、ルーターのアドレスファミリを inet6 と指定することによって VRRP VNIC または L3 VRRP ルーターを作成している必要があります。

VRRP ルーターの仮想 IP アドレスを構成するには、次の構文を使用します。

```
# ipadm create-addr [-t] -T vrrp [-a local=addr[/prefix-length]] \
 [-n router-name].... addr-obj | interface
```

-t　構成されたアドレスは一時的なもので、変更はアクティブな構成だけに適用されることを示します。

-T vrrp　構成されたアドレスのタイプが vrrp であることを指定します。

-n *router-name*　VRRP ルーター名は、IP アドレスが構成されている VRRP VNIC インタフェースから派生できるため、L2 VRRP ルーターでは -n *router-name* オプションを省略できます。

詳細は、ipadm(1M) のマニュアルページを参照してください。

注記 - vrrpadm コマンドの -a オプションを使用して、仮想 IP アドレスを構成することもできます。詳細は、40 ページの「VRRP ルーターの作成」を参照してください。

例 4-3 L2 VRRP ルーターの仮想 IP アドレスの構成

vrrp タイプの IP アドレスを使用して、L2 VRRP ルーターの仮想 IP アドレスを構成できます。次の例は、l2router1 の仮想 IP アドレスの作成方法を示しています。

```
# ipadm create-ip vrrp_vnic1
# ipadm create-addr -T vrrp -n l2router1 -a 192.168.82.8/24 vrrp_vnic1/vaddr1
```

次の例は、V6vrrp_vnic1/vaddr1 の IPv6 リンクローカル vrrp IP アドレスの作成方法を示しています。

```
# ipadm create-ip V6vrrp_vnic1
# ipadm create-addr -T vrrp V6vrrp_vnic1/vaddr1
```

VRRP ルーターの IPv6 リンクローカル vrrp タイプの IP アドレスを構成するために、ローカルアドレスを指定する必要はありません。IPv6 リンクローカル vrrp タイプの IP アドレスは、関連付けられている VRRP ルーターの VRID に基づいて作成されます。

例 4-4 L3 VRRP ルーターの仮想 IP アドレスの構成

次の例は、l3router1 の仮想 IP アドレスの構成方法を示しています。

```
# ipadm create-ip ipmp0
# ipadm create-addr -T vrrp -n l3router1 -a 172.16.82.8/24 ipmp0/vaddr1
```

次の例は、L3 VRRP ルーター l3V6router1 の IPv6 リンクローカル vrrp タイプの IP アドレスの構成方法を示しています。

```
# ipadm create-ip ipmp1
# ipadm create-addr -T vrrp -n l3V6router1 ipmp1/vaddr0
```

VRRP ルーターの有効化および無効化

VRRP ルーターは、最初に作成したときにはデフォルトで有効化されています。VRRP ルーターを無効化して再有効化できます。VRRP ルーターが作成されるインタフェース (vrrpadm

create-router でルーターを作成するときに -l オプションで指定) は、ルーターが有効化されるときに存在する必要があります。それ以外の場合、有効化の操作は失敗します。L2 VRRP ルーターの場合、ルーターの VRRP VNIC が存在しない場合、ルーターは有効ではありません。構文は次のとおりです。

```
# vrrpadm enable-router router-name
```

構成を変更してルーターを再有効化するには、VRRP ルーターの一時的な無効化が必要になる場合があります。ルーターを無効化するための構文は次のとおりです。

```
# vrrpadm disable-router router-name
```

VRRP ルーターの変更

vrrpadm modify-router コマンドは、指定された VRRP ルーターの構成を変更します。ルーターの優先度、通知間隔、プリエンプションモード、および受け入れモードを変更できます。構文は次のとおりです。

```
# vrrpadm modify-router [-p priority] [-i adv-interval] [-o flags] router-name
```

レイヤー 2 およびレイヤー 3 VRRP ルーター構成の表示

vrrpadm show-router コマンドは、指定された VRRP ルーターの構成とステータスを表示します。詳細は、vrrpadm(1M) のマニュアルページを参照してください。構文は次のとおりです。

```
# vrrpadm show-router [-P | -x] [-p] [-o field[,...]] [router-name]
```

例 4-5 レイヤー 2 VRRP ルーター構成の表示

次の例は、vrrpadm show-router コマンド出力を示しています。

```
# vrrpadm show-router vrrp1
NAME VRID TYPE  IFNAME  AF   PRIO ADV_INTV MODE  STATE  VNIC
vrrp1 1   L2    net1    IPv4 100  1000     e-pa- BACK   vnic1
```

NAME　　VRRP ルーターの名前。

VRID　　VRRP ルーターの VRID。

TYPE　　VRRP ルーターのタイプ (L2 または L3)。

IFNAME　VRRP ルーターが構成されるインタフェース。L2 VRRP ルーターの場合、インタフェースは物理 Ethernet インタフェース、VLAN、またはアグリゲーションにできます。

AF　VRRP ルーターのアドレスファミリ。IPv4 または IPv6 のいずれかにできます。

PRIO　VRRP ルーターの、マスター選択で使用される優先度。

ADV_INTV　通知間隔 (ミリ秒で表示)。

MODE　VRRP ルーターに関連付けられるフラグのセットで、次の値を含みます。

- e – ルーターが有効化されていることを指定します。
- p – モードが preempt であることを指定します。
- a – モードが accept であることを指定します。
- o – ルーターが仮想アドレスの所有者であることを指定します。

STATE　VRRP ルーターの現在の状態。使用可能な値は、INIT (初期化)、BACK (バックアップ)、および MAST (マスター) です。

この例では、指定された VRRP ルーター vrrp1 の情報が表示されます。

```
# vrrpadm show-router -x vrrp1
NAME  STATE PRV_STAT STAT_LAST VNIC  PRIMARY_IP VIRTUAL_IPS
vrrp1 BACK  MAST     1m17s     vnic1 10.0.0.100 10.0.0.1
```

PRV_STAT　VRRP ルーターの以前の状態。

STAT_LAST　前回の状態遷移からの時間。

PRIMARY_IP　VRRP ルーターによって選択されたプライマリ IP アドレス。

VIRTUAL_IPS　VRRP ルーター上で構成されている仮想 IP アドレス。

この例では、VRRP ルーターによって選択されるプライマリ IP アドレス、VRRP ルーターに構成される仮想 IP アドレス、VRRP ルーターの前の状態などの追加情報が表示されます。

```
# vrrpadm show-router -P vrrp1
NAME  PEER       P_PRIO  P_INTV  P_ADV_LAST  M_DOWN_INTV
vrrp1 10.0.0.123 120     1000    0.313s      3609
```

PEER　ピア VRRP ルーターのプライマリ IP アドレス。

P_PRIO　ピアから受信した通知の一部のピア VRRP ルーターの優先度。

P_INTV　　ピアから受信した通知の一部の通知間隔 (ミリ秒)。

P_ADV_LAST　　前回ピアから通知を受信してからの時間。

M_DOWN_INTV　　マスタールーターが切断と宣言されてからの時間間隔 (ミリ秒)。

-P オプションが使用されるのは、VRRP ルーターがバックアップ状態になっている場合のみです。

例 4-6　　システム上の L3 VRRP ルーターの表示

```
# vrrpadm show-router
NAME   VRID  TYPE  IFNAME  AF    PRIO  ADV_INTV  MODE   STATE   VNIC
l3vr1  12    L3    net1    IPv6  255   1000      eopa-  INIT    –
```

この例では、L3 VRRP ルーター l3vr1 はインタフェース net1 経由で構成されます。

VRRP ルーターに関連付けられている IP アドレスの表示

ipadm show-addr コマンドを使用して、VRRP ルーターに関連付けられている IP アドレスを表示できます。ipadm show-addr コマンドの出力の ROUTER フィールドには、特定の vrrp タイプの IP アドレスに関連付けられている VRRP ルーターの名前が表示されます。

L2 VRRP の vrrp タイプの IP アドレスの場合、VRRP ルーターの名前は、IP アドレスが構成される VRRP VNIC から派生されます。VRRP VNIC の L2 ルーターの作成前に ipadm show-addr コマンドを発行すると、ROUTER フィールドには ? が表示されます。L3 VRRP の vrrp タイプの IP アドレスの場合は、指定されたルーター名が ROUTER フィールドに常に表示されます。その他のタイプの IP アドレスの場合は、ROUTER フィールドは適用されず、-- が表示されます。

例 4-7　　VRRP ルーターに関連付けられている IP アドレスの表示

```
# ipadm show-addr -o addrobj,type,vrrp-router,addr
ADDROBJ              TYPE     VRRP-ROUTER   ADDR
lo0/v4               static   --               127.0.0.1/8
net1/p1              static   --               192.168.11.10/24
net1/v1              vrrp     l3router1        192.168.81.8/24
vrrp_vnic1/vaddr1    vrrp     l2router1        192.168.82.8/24
lo0/v6               static   --               ::1/128
```

この例では、l3router1 は vrrp タイプの IP アドレス 192.168.81.8/24、l2router1 は vrrp タイプの IP アドレス 192.168.82.8/24 に関連付けられます。

出力には次の情報が表示されます。

ADDROBJ　アドレスオブジェクトの名前。

TYPE　アドレスオブジェクトのタイプで、次のいずれかになります。

- from-gz
- static
- dhcp
- addrconf
- vrrp

VRRP-ROUTER　VRRP ルーターの名前。

ADDR　数値 IPv4 または IPv6 アドレス。

VRRP ルーターの削除

vrrpadm delete-router コマンドは、指定された VRRP ルーターを削除します。構文は次のとおりです。

```
# vrrpadm delete-router router-name
```

注記 - vrrpadm create-router コマンドの -f、-a、-P オプションを使用してそれぞれ作成される VRRP VNIC、vrrp タイプの IP アドレス、およびプライマリ IP アドレスは、vrrpadm delete-router コマンドを実行しても削除されません。対応する ipadm および dladm コマンドを使用して明示的に削除する必要があります。

Gratuitous ARP および NDP メッセージの制御

バックアップルーターがマスター VRRP ルーターになると、VRRP は、マスタールーターに関連付けられるすべての仮想 IP アドレスにフラグを設定するため、仮想 IP アドレスが保護されます。仮想 IP アドレスに競合がない場合は、複数の Gratuitous ARP および近傍通知メッセージ

が送信され、新しいマスターの仮想 IP アドレスと MAC アドレス間のマッピングが通知されます。

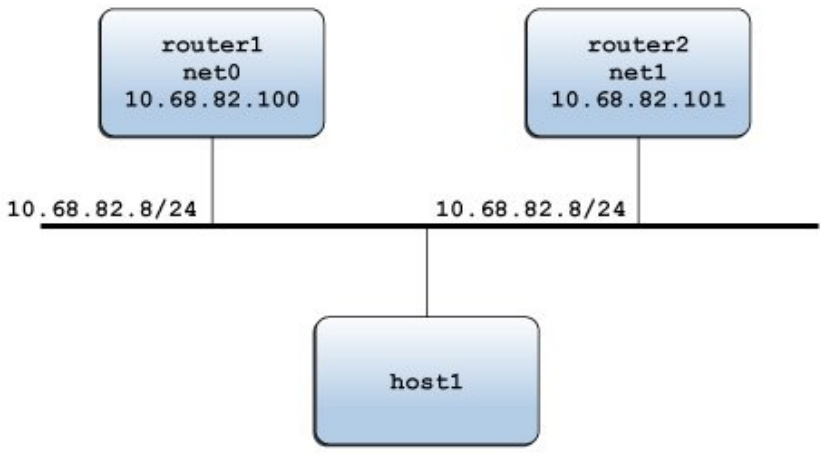
送信されるメッセージの数とメッセージの通知間隔を制御するには、次の IP プロトコルプロパティーを使用できます。

- `arp_publish_count`
- `arp_publish_interval`
- `ndp_unsolicit_count`
- `ndp_unsolicit_interval`

IP プロトコルのプロパティーの詳細は、『Oracle Solaris 11.2 カーネルのチューンアップ・リファレンスマニュアル 』の「重複アドレスの検出に関連した IP チューニング可能パラメータ」を参照してください。

ユースケース: レイヤー 2 VRRP ルーターの構成

次の図は、VRRP の典型的な構成を示したものです。



この例では、IP アドレス `10.68.82.8` が `host1` のデフォルトゲートウェイとして構成されます。この IP アドレスは、2 つの VRRP ルーター `router1` と `router2` から成る仮想ルーターによって保

護される仮想 IP アドレスです。特定の時間に、2 つのルーターのいずれか一方のみがマスタールーターとして機能し、仮想ルーターおよび host1 からの転送パケットの役割を果たすと想定します。

仮想ルーターの VRID が 12 であると想定します。次の例は、router1 と router2 で例の VRRP 構成を構成するために使用されるコマンドを示しています。router1 は仮想 IP アドレス 10.68.82.8 の所有者であり、その優先順位はデフォルト値 (255) です。router2 はスタンバイルーターであり、その優先順位は 100 です。

VRRP の構成に使用されるコマンドの詳細は、vrrpadm(1M)、dladm(1M)、および ipadm(1M) のマニュアルページを参照してください。

router1 の場合:

1. VRRP パッケージをインストールします。

```
# pkg install vrrp
```

2. VRID 値が 12 の net0 上で VNIC vnic0 を作成します。

```
# dladm create-vnic -m vrrp -V 12 -A inet -l net0 vnic0
```

3. net0 上で VRRP ルーター vrrp1 を作成します。

```
# vrrpadm create-router -V 12 -A inet -I net0 vrrp1
```

4. IP インタフェース vnic0 および net0 を構成します。

```
# ipadm create-ip vnic0

# ipadm create-addr -T vrrp -a 10.68.82.8/24 vnic0/router1

# ipadm create-ip net0

# ipadm create-addr -T static -a 10.68.82.100/24 net0/router1
```

5. vrrp1 のルーター情報を表示します。

```
# vrrpadm show-router -x vrrp1
NAME  STATE   PRV_STAT STAT_LAST  VNIC  PRIMARY_IP    VIRTUAL_IPS
vrrp1 MASTER  INIT     14.444s    vnic0 10.68.82.100  10.68.82.8
```

router2 も同様です。

1. VRID 値が 12 の net1 上で VNIC vnic1 を作成します。

```
# dladm create-vnic -m vrrp -V 12 -A inet -l net1 vnic1
```

2. net1 上で VRRP ルーター vrrp2 を作成します。

```
# vrrpadm create-router -V 12 -A inet -I net1 -p 100 vrrp2
```

3. vnic1 および net1 上の IP インタフェースを構成します。

```
# ipadm create-ip vnic1

# ipadm create-addr -T vrrp -a 10.68.82.8/24 vnic1/router2

# ipadm create-ip net1

# ipadm create-addr -T static -a 10.68.82.101/24 net1/router2
```

4. vrrp2 のルーター情報を表示します。

```
# vrrpadm show-router -x vrrp2
NAME  STATE    PRV_STAT  STAT_LAST  VNIC  PRIMARY_IP    VIRTUAL_IPS
vrrp2 BACKUP   INIT      2m32s      vnic1 10.68.82.101  10.68.82.8
```

router1 の構成を例として使用して、net0 上で IP アドレスを少なくとも 1 つ構成する必要があります。router1 のこの IP アドレスは、VRRP 通知パケットの送信に使用されるプライマリ IP アドレスです。

```
# vrrpadm show-router -x vrrp1
NAME  STATE   PRV_STAT STAT_LAST  VNIC  PRIMARY_IP    VIRTUAL_IPS
vrrp1 MASTER  INIT     14.444s    vnic1 10.68.82.100  10.68.82.8
```

♦ ♦ ♦ 第 5 章

5 統合ロードバランサの概要

この章では、ILB コンポーネントおよび ILB の動作モード (Direct Server Return (DSR) モードやネットワークアドレス変換 (NAT) モードなど) のコンポーネントについて説明します。

この章の内容は、次のとおりです。

ILB の詳細は、13 ページの「統合ロードバランサの概要」を参照してください。

ILB のコンポーネント

ILB はサービス管理機能 (SMF) サービス `svc:/network/loadbalancer/ilb:default` によって管理されています。SMF の詳細は、『Oracle Solaris 11.2 でのシステムサービスの管理 』を参照してください。ILB の 3 つの主要なコンポーネント:

- `ilbadm` コマンド行インタフェース (CLI) – この CLI を使用して、負荷分散規則の構成、オプションの健全性検査の実行、および統計の表示が可能です。
- `libilb` 構成ライブラリ – `ilbadm` およびサードパーティーアプリケーションは、`libilb` 内に実装されている ILB 管理用の機能を使用できます。
- `ilbd` デーモン – このデーモンは次のタスクを実行します。
  - リブート後やパッケージ更新後にも持続する構成を管理します。
  - 構成情報を処理し、その情報を ILB カーネルモジュールに送信して実行することによって、ILB カーネルモジュールへの逐次アクセスを提供します
  - 健全性検査を実行し、結果を ILB カーネルモジュールに送信することで、負荷分散が正しく調整されます

ILB の動作モード

ILB は、1 脚または 2 脚のトポロジで、IPv4 および IPv6 に対するステートレス DSR および NAT の動作モードをサポートします。

Direct Server Return モード

DSR モードでは、ILB は受信リクエストをバックエンドサーバーに分散しますが、サーバーからクライアントへの戻りトラフィックは ILB をバイパスします。ただし、ILB がバックエンドサーバーのルーターとして使用されるように設定すると、バックエンドサーバーからクライアントへの応答は、ILB を実行しているシステムを通るようにルーティングされます。ILB の現在の DSR 実装は TCP 接続追跡を提供せず、ステートレスにします。ステートレス DSR では、ILB は基本的な統計情報を除き、処理されるパケットのステート情報を保存しません。ステートレス中、パフォーマンスは通常の IP 転送のパフォーマンスに相当します。DSR モードはコネクションレスプロトコルに最適です。

メリット:

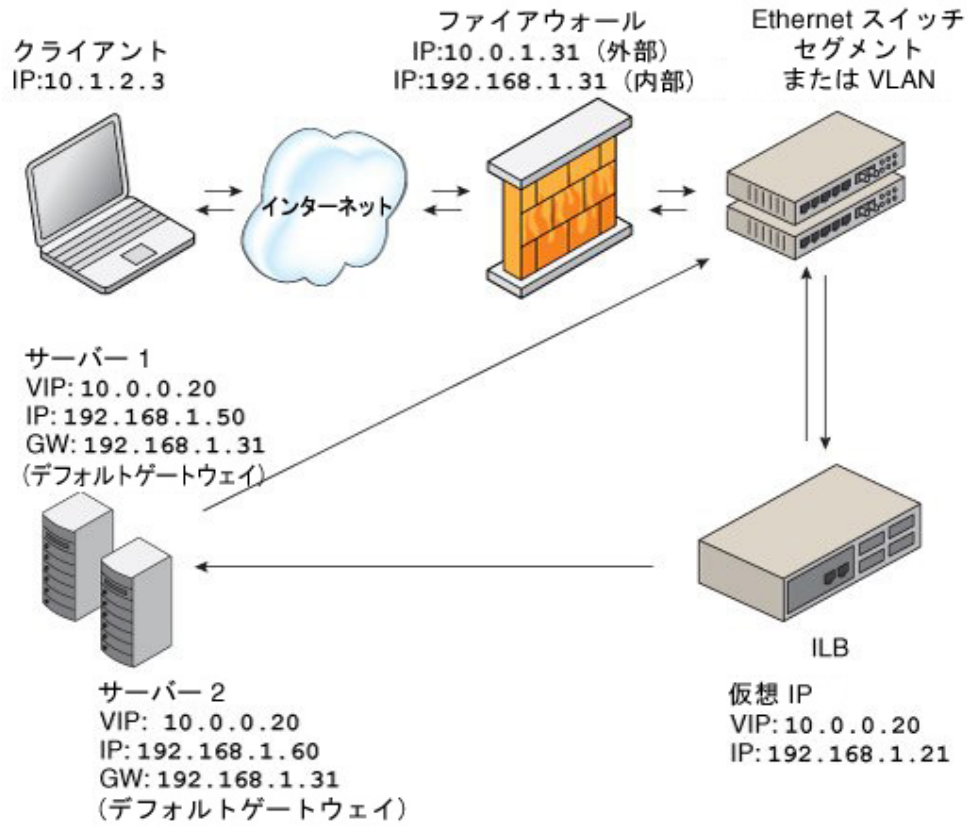
- DSR では、パケットの着信先 MAC アドレスのみが変更され、サーバーがクライアントに直接応答するため、NAT よりもパフォーマンスが優れています。
- サーバーとクライアントの間に完全な透過性があります。サーバーはクライアント IP アドレスから接続を直接認識し、デフォルトゲートウェイを介してクライアントに応答します。

デメリット:

- バックエンドサーバーは、それ固有の IP アドレス (健全性検査用) および仮想 IP アドレス (負荷分散トラフィック用) の両方に応答する必要があります。
- ステートレス中に、サーバーを追加または削除すると、接続の中断が発生します。

次の図に、DSR モードでの ILB の実装を示します。

図 5-1 Direct Server Return トポロジ



この図で、バックエンドサーバーはどちらも ILB ボックスと同じサブネット (192.168.1.0/24) 内にあります。また、サーバーはルーターにも接続されているため、ILB ボックスから転送されたリクエストを受け取ったあと、クライアントに直接応答できます。

ネットワークアドレス変換モード

ILB は負荷分散機能のためだけに、NAT をスタンドアロンモードで使用します。このモードでは、ILB はヘッダー情報を書き換え、受信トラフィックと送信トラフィックを処理します。ILB はハーフ NAT モードとフル NAT モードの両方で動作します。ただし、フル NAT は発信元 IP

アドレスも書き換えるため、サーバーには、すべての接続がロードバランサから発信されているように見えます。NAT は TCP 接続追跡を提供します (つまりステートフルです)。NAT モードは、追加のセキュリティーを提供し、ハイパーテキスト転送プロトコル (HTTP) または Secure Sockets Layer (SSL) トラフィックに最適です。

メリット:

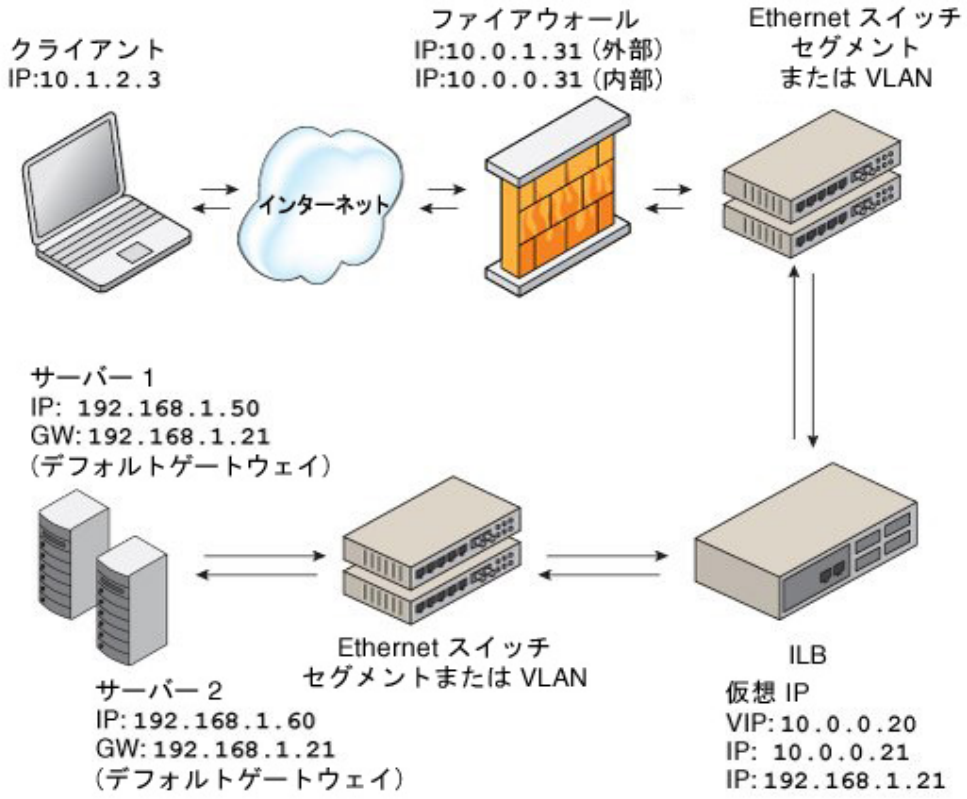
- ロードバランサを指定するようにデフォルトゲートウェイを変更することによって、すべてのバックエンドサーバーで動作します。
- ロードバランサにより接続状態が保持されるため、接続を中断せずにサーバーを追加または削除できます。

デメリット:

- 処理に IP ヘッダーの操作を伴い、サーバーがロードバランサに応答を送信するため、DSR よりもパフォーマンスが低くなります。
- すべてのバックエンドサーバーがロードバランサをデフォルトゲートウェイとして使用する必要があります。

NAT モードの一般的な実装を、次の図に示します。

図 5-2 ネットワークアドレス変換トポロジ



この場合、VIP へのリクエストはすべて ILB を経由し、バックエンドサーバーに転送されます。バックエンドサーバーからの応答はすべて、NAT のために ILB を経由します。

注意 - ILB に実装されている NAT コードパスは、Oracle Solaris の IP フィルタ機能に実装されているコードパスとは異なります。これらのコードパスを同時に使用*しないでください*。

ハーフ NAT 負荷分散モード

ILB のハーフ NAT 動作モードでは、ILB はパケットのヘッダー内の着信先 IP アドレスのみを書き換えます。ハーフ NAT 実装を使用している場合、サーバーが存在する同一のサブネット

から、サービスの VIP アドレスに接続できません。次の表に、クライアントと ILB の間、および ILB とバックエンドサーバーの間を流れるパケットの IP アドレスを示します。

表 5-1 サーバーとクライアントが異なるネットワーク上にある場合のハーフ NAT 実装のリクエストフローと応答フロー

	リクエストフロー	発信元 IP アドレス	着信先 IP アドレス
1.	クライアント –> ILB	クライアント	ILB の VIP
2.	ILB –> サーバー	クライアント	サーバー
	応答フロー		
3.	サーバー –> ILB	サーバー	クライアント
4.	ILB –> クライアント	ILB の VIP	クライアント

クライアントシステムをサーバーと同じネットワークに接続した場合、意図したサーバーはクライアントに直接応答し、表の 4 番目の手順は発生しません。したがって、クライアントに対するサーバー応答の発信元 IP アドレスは無効です。クライアントが接続リクエストをロードバランサに送信すると、応答は意図したサーバーから発生します。これ以降のクライアントの IP スタックはすべての応答を適切に削除します。このシナリオでは、リクエストフローと応答フローは次の表に示すとおりに進行します。

表 5-2 サーバーとクライアントが同じネットワーク上にある場合のハーフ NAT 実装のリクエストフローと応答フロー

	リクエストフロー	発信元 IP アドレス	着信先 IP アドレス
1.	クライアント –> ILB	クライアント	ILB の VIP
2.	ILB –> サーバー	クライアント	サーバー
	応答フロー		
3.	サーバー –> クライアント	サーバー	クライアント

フル NAT 負荷分散モード

ILB 動作のフル NAT 実装では、発信元 IP アドレスと着信先 IP アドレスが書き換えられることで、トラフィックがロードバランサを両方向で通過します。フル NAT モードでは、サーバーが存在する同一のサブネットから VIP に接続できるようになります。

次の表に、フル NAT モードの使用時にクライアントと ILB の間、および ILB とバックエンドサーバーの間を流れるパケットの IP アドレスを示します。サーバーには、ILB ボックスを使用す

る特別なデフォルトルートは不要です。ただし、フル NAT モードでは、ILB がバックエンドサーバーとの通信に発信元アドレスとして使用する 1 つの IP アドレスまたは IP アドレス範囲を、管理者が別途設定する必要があります。使用するアドレスがサブネット C に属しているとします。このシナリオでは、ILB はプロキシとして動作します。

表 5-3 フル NAT 実装のリクエストフローと応答フロー

	リクエストフロー	発信元 IP アドレス	着信先 IP アドレス
1.	クライアント –> ILB	クライアント	ILB の VIP
2.	ILB –> サーバー	ロードバランサのインタフェースアドレス (サブネット C)	サーバー
	応答フロー		
3.	サーバー –> ILB	サーバー	ILB のインタフェースアドレス (サブネット C)
4.	ILB –> クライアント	ILB の VIP	クライアント

ILB の動作

このセクションでは、クライアントから VIP へのリクエストの処理、バックエンドサーバーへのリクエストの転送、および応答の処理を伴う、ILB のプロセスについて説明します。

クライアントからサーバーへのパケット処理の手順は次のとおりです。

1. ILB は、クライアントから VIP アドレスに送信された受信リクエストを受け取り、リクエストを負荷分散規則と照合します。
2. ILB は一致する負荷分散規則を見つけると、負荷分散アルゴリズムを使用して、動作モードに応じてリクエストをバックエンドサーバーに転送します。
   - DSR モードでは、ILB は受信リクエストの MAC ヘッダーを、選択されたバックエンドサーバーの MAC ヘッダーに置換します。
   - ハーフ NAT モードでは、ILB は受信リクエストの着信先 IP アドレスおよびトランスポートプロトコルのポート番号を、選択されたバックエンドサーバーのものに置換します。
   - フル NAT モードでは、ILB は受信リクエストの発信元 IP アドレスとトランスポートプロトコルのポート番号を、負荷分散規則の NAT 発信元アドレスに置換します。また、ILB は受信リクエストの着信先 IP アドレスおよびトランスポートプロトコルのポート番号を、選択されたバックエンドサーバーのものに置換します。
3. ILB は変更された受信リクエストを、選択されたバックエンドサーバーに転送します。

サーバーからクライアントへのパケット処理の手順は次のとおりです。

1. バックエンドサーバーはクライアントからの受信リクエストに応答して、返信を ILB に送信します。
2. バックエンドサーバーから応答を受け取ったあとの ILB のアクションは、動作モードに基づきます。
   - DSR モードでは、バックエンドサーバーからの応答は ILB をバイパスし、クライアントに直接届きます。ただし、ILB がバックエンドサーバーのルーターとしても使用される場合、バックエンドサーバーからクライアントへの応答は、ILB を実行しているシステムを通るようにルーティングされます。
   - ハーフ NAT モードとフル NAT モードでは、ILB はバックエンドサーバーからの応答を受信リクエストと照合し、変更された IP アドレスおよびトランスポートプロトコルのポート番号を元の受信リクエストのものに置換します。その後、ILB はクライアントに応答を転送します。

♦ ♦ ♦ 第 6 章

6 統合ロードバランサの構成と管理

ILB は、ネットワークのプロトコルスタックの L3 および L4 レイヤー上で構成されます。この章では、ILB のインストール、ILB の有効化または無効化、ILB のサーバーグループとバックエンドサーバーの定義、および `ilbadm` コマンドを使用した ILB 構成のエクスポートとインポートのタスクについて説明します。詳細は、`ilbadm`(1M) のマニュアルページを参照してください。ILB の高可用性の構成の詳細は、第7章「ILB の高可用性の構成」を参照してください。

Oracle Solaris 統合ロードバランサの配備方法の詳細は、Deploying the Oracle Solaris Integrated Load Balancer in 60 Minutes (http://www.oracle.com/technetwork/systems/hands-on-labs/hol-deploy-ilb-60mmin-2137812.html) を参照してください。Oracle Solaris Zones および Oracle Solaris の ILB を使用して高可用性をアプリケーションに追加する方法の詳細は、How to Set Up a Load-Balanced Application Across Two Oracle Solaris Zones (http://www.oracle.com/technetwork/articles/servers-storage-admin/loadbalancedapp-1653020.html) を参照してください。

この章の内容は、次のとおりです。

ILB のインストール

ILB にはカーネルとユーザーランドのインストールがあります。ILB のカーネルインストールは、Oracle Solaris のインストールの一環として自動的に実行されます。ただし、次のコマンドを使用して ILB のユーザーランドインストールを実行する必要があります。

```
# pkg install ilb
```

コマンド行インタフェースを使用した ILB の構成

ILB CLI は `/usr/sbin/ilbadm` ディレクトリにあります。CLI には、負荷分散規則、サーバーグループ、および健全性検査を構成するサブコマンドが含まれています。また、統計情報や構成の詳細を表示するサブコマンドも含まれています。`ilbadm show-rule`、`ilbadm show-server`、`ilbadm show-healthcheck` などのサブコマンドの表示を除く、ILB 構成サブコマンドのユーザー承認を設定する必要があります。ILB 構成サブコマンドを実行するには、`solaris.network.ilb.config` RBAC 承認が必要です。

- 既存のユーザーに承認を割り当てる方法を見つけるには、『Oracle Solaris 11.2 でのユーザーとプロセスのセキュリティー保護 』の第 3 章「Oracle Solaris での権利の割り当て」を参照してください。
- システムに新規ユーザーアカウントを作成するときも承認を付与できます。

  次の例では、ユーザー `ilbadm` をグループ ID `10`、ユーザー ID `1210` で作成し、システムの ILB を管理する承認を与えます。

  ```
  # useradd -g 10 -u 1210 -A solaris.network.ilb.config ilbadmin
  ```

  `useradd` コマンドは、`/etc/passwd`、`/etc/shadow`、および `/etc/user_attr` ファイルに新しいユーザーを追加します。`-A` オプションは、ユーザーに承認を割り当てます。

サブコマンドは、次の 2 つのカテゴリに分けられます。

- **構成サブコマンド** – これらのサブコマンドでは次のタスクを実行できます。
  - 負荷分散規則の作成および削除
  - 負荷分散規則の有効化および無効化
  - サーバーグループの作成および削除
  - サーバーグループへのサーバーの追加および削除

- バックエンドサーバーの有効化および無効化
- 負荷分散規則内のサーバーグループに関するサーバー健全性検査の作成および削除

注記 - 構成サブコマンドを管理する特権が必要です。適切な役割を作成してユーザーに割り当てるには、『Oracle Solaris 11.2 でのユーザーとプロセスのセキュリティー保護 』の「役割の作成」を参照してください。

- **表示サブコマンド** – これらのサブコマンドでは次のタスクを実行できます。
  - 構成された負荷分散規則、サーバーグループ、および健全性検査の表示
  - パケット転送統計の表示
  - NAT 接続テーブルの表示
  - 健全性検査結果の表示
  - セッション持続性マッピングテーブルの表示

注記 - 表示サブコマンドを管理するために特権は不要です。

`ilbadm` サブコマンドの詳細は、`ilbadm`(1M) のマニュアルページを参照してください。

ILB の有効化または無効化

このセクションでは、ILB のインストール後に ILB を有効化する方法、または ILB サービスが必要ない場合に ILB を無効化する方法について説明します。

▼ ILB を有効にする方法

1. **管理者になります。**
   詳細は、『Oracle Solaris 11.2 でのユーザーとプロセスのセキュリティー保護 』の「割り当てられている管理権利の使用」を参照してください。

2. **適切な転送サービス (IPv4 と IPv6 のいずれか、またはその両方) を有効にします。**
   このコマンドは正常に終了した場合は出力を生成しません。

```
# ipadm set-prop -p forwarding=on ipv4
# ipadm set-prop -p forwarding=on ipv6
```

3. **ILB サービスを有効にします。**

```
# svcadm enable ilb
```

4. **ILB サービスが有効になっていることを確認します。**

```
# svcs ilb
```

このコマンドは、サービス構成リポジトリに記録されているサービスインスタンスに関する情報を表示します。

▼ ILB を無効にする方法

ILB サービスが不要な場合は、ILB を無効化できます。

1. **管理者になります。**

詳細は、『Oracle Solaris 11.2 でのユーザーとプロセスのセキュリティー保護 』の「割り当てられている管理権利の使用」を参照してください。

2. **ILB サービスを無効にします。**

```
# svcadm disable ilb
```

3. **ILB サービスが無効になっていることを確認します。**

```
# svcs ilb
```

このコマンドは、サービス構成リポジトリに記録されているサービスインスタンスに関する情報を表示します。

次の手順 ILB サービスが無効化され、必要ない場合は、IP 転送を無効にする必要があります。

ILB の管理

ILB の設定 (サーバーグループの定義、ILB の健全性検査のモニタリング、および ILB の有効化後の ILB 規則の作成) が可能です。

このセクションの内容は次のとおりです。

ILB のサーバーグループおよびバックエンドサーバーの定義

このセクションでは、ILB サーバーグループを作成して、バックエンドサーバーをサーバーグループに追加する方法について説明します。`create-servergroup` サブコマンドまたは `add-server` サブコマンドのいずれかを使用してサーバーが追加されると、サーバー ID がシステムによって生成されます。サーバー ID はサーバーグループ内で一意です。`ilbadm` サブコマンドの詳細は、`ilbadm`(1M) のマニュアルページを参照してください。

ILB サーバーグループの作成

ILB サーバーグループを作成するには、まずサーバーグループに含めるサーバーを特定します。サーバーは、ホスト名または IP アドレスと、オプションのポートによって指定できます。次に、管理者として次のコマンドを実行します。

```
# ilbadm create-servergroup -s servers=server1,server2,server3  servergroup
```

追加されるサーバーごとに、先頭に下線 (_) が付いた一意のサーバーID が生成されます。

注記 - サーバーが複数のサーバーグループに属する場合は、複数のサーバー ID を持つことができます。

ILB サーバーグループへのバックエンドサーバーの追加

バックエンドサーバーをサーバーグループに追加するには、管理者になり次のコマンドを実行します。

```
# ilbadm add-server -s server=server1[,server2...] servergroup
```

サーバー指定にはホスト名または IP アドレスを含める必要があり、オプションのポートまたはポート範囲を含めることもできます。同一の IP アドレスを持つサーバーエントリは、1 つのサー

バーグループ内で許可されません。追加されるサーバーごとに、先頭に下線 (_) が付いた一意のサーバーID が生成されます。

注記 - IPv6 アドレスは、角括弧で囲む必要があります。

例 6-1 ILB サーバーグループの作成およびバックエンドサーバーの追加

次の例では、3 つのバックエンドサーバーが含まれる webgroup というサーバーグループを作成します。

```
# ilbadm create-servergroup -s \
servers=192.168.89.11,192.168.89.12,192.168.89.13 webgroup
# ilbadm show-servergroup
SGNAME         SERVERID            MINPORT MAXPORT IP_ADDRESS
webgroup       _webgroup.0         --      --      192.168.89.11
webgroup       _webgroup.1         --      --      192.168.89.12
webgroup       _webgroup.2         --      --      192.168.89.13
```

次の例では、webgroup1 というサーバーグループを作成し、3 つのバックエンドサーバーをサーバーグループに追加します。

```
# ilbadm create-servergroup webgroup1
# ilbadm add-server -s server=[2001:0db8:7::feed:6]:8080,\
[2001:0db8:7::feed:7]:8080,[2001:0db8:7::feed:8]:8080 webgroup1
```

ILB サーバーグループのバックエンドサーバーの有効化または無効化

最初に、再有効化または無効化するバックエンドサーバーの IP アドレス、ホスト名、またはサーバー ID を特定します。サーバーグループ内のサーバーを有効または無効化する前に、サーバーグループを規則に関連付ける必要があります。

サーバーが複数のサーバーグループに属する場合は、複数のサーバー ID を持つことができます。サーバー ID に関連付けられている特定の規則でサーバーを再有効化または無効化するには、そのサーバー ID を指定する必要があります。

- 有効化されたサーバーを無効化するには、次のコマンドを入力します。

  # ilbadm disable-server *server1*

  選択されたサーバーは有効化されていますが、無効化されます。カーネルはトラフィックをこのサーバーに転送しません。

- 無効化されたサーバーを再有効化するには、次のコマンドを入力します。

  ```
  # ilbadm enable-server server1
  ```

  選択されたサーバーは無効にされていますが、再有効化されます。
- サーバーの状態を表示するには、次のコマンドを入力します。

  ```
  # ilbadm show-server [[-p] -o field[,field...]] [rulename]
  ```

注記 - サーバーには、サーバーが属するサーバーグループが規則に関連付けられている場合にのみ、有効化または無効化の状態が表示されます。

例 6-2 ILB サーバーグループのバックエンドサーバーの無効化および再有効化

次の例では、サーバー ID `_websg.1` のサーバーが無効化され、その後再有効化されます。

```
# ilbadm enable-server _websg.1
# ilbadm disable-server _websg.1
```

ILB サーバーグループからのバックエンドサーバーの削除

`ilbadm remove-server` コマンドを使用して、1 つの ILB サーバーグループまたはすべてのサーバーグループからバックエンドサーバーを削除できます。最初に、サーバーグループから削除するサーバーのサーバー ID を特定します。

```
ilbadm show-servergroup -o all
```

サーバー ID は、サーバーがサーバーグループに追加されたときにシステムに割り当てられる IP アドレスに対応する一意の名前です。

次に、サーバーを削除します。

```
# ilbadm remove-server -s server=server-ID server-group
```

サーバーが NAT またはハーフ NAT 規則によって使用されている場合は、削除の前に `disable-server` サブコマンドを使用してサーバーを無効にしてください。詳細は、64 ページの「ILB サーバーグループのバックエンドサーバーの有効化または無効化」を参照してください。サーバーが無効になると、サーバーは接続排出状態に入ります。`ilbadm show-nat` コマンドを使用して、NAT テーブルを定期的にチェックし、サーバーにまだ接続がある

かどうかを確認します。すべての接続が排出されたら (サーバーは show-nat コマンド出力に表示されません)、サーバーは remove-server コマンドを使用して削除できます。

conn-drain タイムアウト値が設定されている場合、接続排出状態はタイムアウト期間が終了した時点で完了します。conn-drain タイムアウトのデフォルト値は 0 で、つまり接続が正常にシャットダウンされるまで接続排出が待機し続けることを意味します。

例 6-3 ILB サーバーグループからのバックエンドサーバーの削除

次の例では、サーバー ID _sg1.2 を持つサーバーをサーバーグループ sg1 から削除します。

```
# ilbadm remove-server -s server=_sg1.2 sg1
```

ILB サーバーグループの削除

このセクションでは、ILB サーバーグループの削除方法について説明します。アクティブな規則によって使用されているサーバーグループは削除できません。

最初に、サーバーグループに関する使用可能なすべての情報を表示します。

```
# ilbadm show-servergroup -o all
sgname      serverID       minport     maxport     IP_address
specgroup   _specgroup.0   7001        7001        192.168.68.18
specgroup   _specgroup.1   7001        7001        192.168.68.19
test123     _test123.0     7002        7002        192.168.67.18
test123     _test123.1     7002        7002        192.168.67.19
```

次のコマンドを入力します。

```
# ilbadm delete-servergroup servergroup
```

サーバーグループがアクティブな規則で使用されている場合は、削除に失敗します。

次の例では、webgroup というサーバーグループを削除します。

```
# ilbadm delete-servergroup webgroup
```

ILB の健全性検査のモニタリング

ILB では、次のオプションのタイプのサーバー健全性検査が提供されています。

- 組み込み ping プローブ

- 組み込み TCP プローブ
- 組み込み UDP プローブ
- 健全性検査として実行できるユーザー指定のカスタムテスト

デフォルトでは、ILB は健全性検査を実行しません。負荷分散規則を作成するとき、サーバーグループごとに健全性検査を指定できます。1 つの負荷分散規則につき 1 つの健全性検査のみを構成できます。仮想サービスが有効であるかぎり、有効化されている仮想サービスに関連付けられたサーバーグループの健全性検査は自動的に開始され、定期的に繰り返されます。仮想サービスが無効化されると健全性検査はすぐに停止します。仮想サービスがふたたび有効化されたとき、以前の健全性検査状態は保持されていません。

健全性検査を実行するために TCP プローブ、UDP プローブ、またはカスタムテストプローブを指定した場合、ILB はデフォルトで、指定された TCP プローブ、UDP プローブ、またはカスタムテストプローブをサーバーに送信する前に、サーバーが到達可能かどうかを判別するために ping プローブを送信します。ping プローブに失敗すると、対応するサーバーは、健全性検査ステータスが `unreachable` になり無効化されます。ping プローブに成功しても、TCP プローブ、UDP プローブ、またはカスタムテストプローブに失敗した場合、サーバーは健全性検査ステータスが `dead` になり無効化されます。

UDP プローブ以外のデフォルトの ping プローブを無効化できます。ping プローブは、常に UDP 健全性検査のデフォルトのプローブになります。

健全性検査の作成

負荷分散規則を作成する場合は、健全性検査を作成してサーバーグループに割り当てることができます。次の例では、`hc1` と `hc-myscript` の 2 つの健全性検査オブジェクトが作成されます。最初の健全性検査は組み込み TCP プローブを使用します。2 番目の健全性検査はカスタムテスト `/var/tmp/my-script` を使用します。

```
# ilbadm create-healthcheck -h hc-timeout=3,\
hc-count=2,hc-interval=8,hc-test=tcp hc1
# ilbadm create-healthcheck -h hc-timeout=3,\
hc-count=2,hc-interval=8,hc-test=/var/tmp/my-script hc-myscript
```

引数は次のとおりです。

`hc-timeout` 健全性検査が完了しない場合に失敗したと見なされるまでのタイムアウトを指定します。

`hc-count` `hc-test` 健全性検査の実行を試行する回数を指定します。

`hc-interval` 連続する健全性検査の間隔を指定します。プローブをすべてのサーバーに同時に送信することを回避するために、実際の間隔は `0.5 * hc-interval` から `1.5 * hc-interval` の間でランダム化されます。

`hc-test` 健全性検査の種類を指定します。`tcp`、`udp`、`ping` などの組み込みの健全性検査、またはフルパス名で指定する必要のある外部の健全性検査を指定できます。

注記 - `hc-test` のポート指定は、`create-rule` サブコマンドの `hc-port` キーワードで指定します。詳細は、`ilbadm`(1M) のマニュアルページを参照してください。

ユーザー指定のカスタムテストはバイナリまたはスクリプトになります。

- テストはシステム上の任意の場所に配置できます。`create-healthcheck` サブコマンドを使用する場合は、絶対パスを指定する必要があります。

  `create-rule` サブコマンドの健全性検査指定の一部としてテスト (`/var/tmp/my-script` など) を指定すると、`ilbd` デーモンがプロセスをフォークし、次のようにテストを実行します。

  ```
  /var/tmp/my-script $1 $2 $3 $4 $5
  ```

  引数は次のとおりです。

  `$1` VIP (リテラルの IPv4 または IPv6 アドレス)

  `$2` サーバー IP (リテラルの IPv4 または IPv6 アドレス)

  `$3` プロトコル (文字列としての UDP、TCP)

  `$4` 数値ポート範囲 (`hc-port` に対するユーザー指定の値)

  `$5` 失敗を返す前にテストが待機する最大時間 (秒)。指定された時間を超えてテストが実行されると、停止される可能性があり、テストは失敗したと見なされます。この値はユーザーによって定義され、`hc-timeout` に指定されます。

- ユーザー指定のテストは、すべての引数を使用しても使用しなくてもかまいませんが、次のいずれかを返す*必要があります*。
  - マイクロ秒単位の往復時間 (RTT)
  - テストが RTT を計算しない場合は 0
  - 失敗した場合は -1

デフォルトでは、健全性検査テストは PRIV_PROC_FORK、RIV_PROC_EXEC、および RIV_NET_ICMPACCESS の特権で実行されます。

さらに広い特権セットが必要な場合、テストで setuid を実装する必要があります。特権の詳細は、privileges(5) のマニュアルページを参照してください。

健全性検査の一覧表示

構成済みの健全性検査の詳細情報を取得するには、次のコマンドを発行します。

```
# ilbadm show-healthcheck
HCNAME      TIMEOUT COUNT   INTERVAL DEF_PING TEST
hc1         3       2       8        Y        tcp
hc2         3       2       8        N        /var/usr-script
```

健全性検査結果の表示

ilbadm list-hc-result コマンドを使用して、健全性検査の結果を取得します。規則または健全性検査を指定しない場合、サブコマンドはすべての健全性検査を一覧表示します。

次の例では、rule1 という規則に関連付けられた健全性検査の結果を表示します。

```
# ilbadm show-hc-result rule1
RULENAME    HCNAME      SERVERID    STATUS    FAIL LAST      NEXT      RTT
rule1       hc1         _sg1:0      dead      10   11:01:19  11:01:27  941
rule1       hc1         _sg1:1      alive     0    11:01:20  11:01:34  1111
```

注記 - show-hc-result コマンドは、関連付けられた健全性検査が規則にある場合のみ、健全性検査の結果を表示します。

出力の LAST 列は、サーバーで健全性検査が実行された時間を示します。NEXT 列は、次の健全性検査が実行される時間を示します。

健全性検査の削除

ilbadm delete-healthcheck コマンドを使用して、健全性検査を削除します。次の例では、hc1 という健全性検査を削除します。

```
# ilbadm delete-healthcheck hc1
```

ILB 規則の構成

このセクションでは、ilbadm コマンドを使用して負荷分散規則を作成、削除、および一覧表示する方法について説明します。

ILB のアルゴリズム

ILB のアルゴリズムはトラフィック分散を制御し、負荷分散およびサーバー選択のためのさまざまな特性を提供します。

ILB は 2 つの動作モードに対して次のアルゴリズムを提供します。

- ラウンドロビン – ラウンドロビンアルゴリズムでは、ロードバランサはサーバーグループに対してローテーションベースでリクエストを割り当てます。サーバーにリクエストが割り当てられると、そのサーバーはリストの末尾に移動します。
- *src-IP* ハッシュ – 発信元 IP ハッシュ方式では、ロードバランサは受信リクエストの発信元 IP アドレスのハッシュ値に基づいてサーバーを選択します。
- *src-IP, port* ハッシュ – 発信元 IP、ポートハッシュ方式では、ロードバランサは受信リクエストの発信元 IP アドレスおよび発信元ポートのハッシュ値に基づいてサーバーを選択します。
- *src-IP, VIP* ハッシュ – 発信元 IP、VIP ハッシュ方式では、ロードバランサは受信リクエストの発信元 IP アドレスおよび着信先 IP アドレスのハッシュ値に基づいてサーバーを選択します。

ILB 規則の作成

ILB では、仮想サービスは負荷分散規則によって表現され、次のパラメータで定義されます。

- 仮想 IP アドレス
- トランスポートプロトコル: TCP または UDP
- ポート番号 (またはポート範囲)
- 負荷分散アルゴリズム
- 負荷分散モード (DSR、フル NAT、またはハーフ NAT)
- 一連のバックエンドサーバーで構成されるサーバーグループ
- サーバーグループ内の各サーバーに対して実行できる、オプションのサーバー健全性検査
- 健全性検査に使用するオプションのポート

注記 - 健全性検査は、特定のポートか、`ilbd` デーモンがサーバーのポート範囲からランダムに選択する任意のポートを指定できます。

- 仮想サービスを表す規則名

規則を作成する前に、次を実行する必要があります。

- 該当するバックエンドサーバーを含むサーバーグループを作成します。詳細は、63 ページの「ILB のサーバーグループおよびバックエンドサーバーの定義」を参照してください。
- 健全性検査を作成し、サーバー健全性検査と規則を関連付けます。詳細は、67 ページの「健全性検査の作成」を参照してください。
- 規則に関連付ける VIP、ポート、およびオプションのプロトコルを特定します。
- 使用する動作 (DSR、ハーフ NAT、またはフル NAT) を特定します。
- 使用する負荷分散アルゴリズムを特定します。詳細は、70 ページの「ILB のアルゴリズム」を参照してください。

`ilbadm create-rule` コマンドを使用して、ILB 規則を作成します。`ilbadm create-rule` コマンドの使用の詳細は、`ilbadm`(1M) のマニュアルページを参照してください。

構文は次のとおりです。

```
# ilbadm create-rule -e -i vip=IPaddr,port=port,protocol=protocol \
-m lbalg=lb-algorithm,type=topology-type,proxy-src=IPaddr1-IPaddr2,\
pmask=value -h hc-name=hc1 -o servergroup=sg rule1
```

注記 - `-e` オプションでは、作成する規則を有効にします (このオプションを使用しない場合は、デフォルトでは無効になります)。

例 6-4 健全性検査セッション持続性を持つフル NAT 規則の作成

この例では、`hc1` という健全性検査と、`sg1` というサーバーグループを作成します。サーバーグループは、それぞれポート範囲を持つ 2 つのサーバーで構成されます。最後のコマンドは、`rule1` という規則を作成して有効にし、この規則をサーバーグループおよび健全性検査に関連付けます。この規則はフル NAT 動作モードを実装します。サーバーグループおよび健全性検査の作成は、規則の作成よりも前に行う必要があります。

```
# ilbadm create-healthcheck -h hc-test=tcp,hc-timeout=2,\
```

```
hc-count=3,hc-interval=10 hc1
# ilbadm create-servergroup -s server=192.168.0.10:6000-6009,192.168.0.11:7000-7009 sg1
# ilbadm create-rule -e -p -i vip=10.0.0.10,port=5000-5009,\
protocol=tcp -m lbalg=rr,type=NAT,proxy-src=192.168.0.101-192.168.0.104,pmask=24 \
-h hc-name=hc1 -o servergroup=sg1 rule1
```

持続性マッピングを作成すると、その後に続く仮想サービスへの接続またはパケット、あるいはその両方のリクエストは、クライアントの発信元 IP アドレスが一致する場合、同一のバックエンドサーバーに転送されます。クラスレスドメイン間ルーティング (CIDR) 表記の接頭辞長は、IPv4 では `0-32`、IPv6 では `0-128` の間の値です。

ハーフ NAT またはフル NAT 規則を作成する場合は、`connection-drain` タイムアウト値を指定します。`conn-drain` タイムアウトのデフォルト値は `0` で、つまり接続が正常にシャットダウンされるまで接続排出が待機し続けることを意味します。

ILB 規則の一覧表示

規則の構成の詳細を一覧表示するには、次のコマンドを発行します。規則名を指定しない場合、すべての規則の情報が提供されます。

```
# ilbadm show-rule
RULENAME        STATUS   LBALG           TYPE    PROTOCOL VIP           PORT
rule-http       E        hash-ip-port    NAT     TCP      10.0.0.1      80
rule-dns        D        hash-ip         NAT     UDP      10.0.0.1      53
rule-abc        D        roundrobin      NAT     TCP      2001:db8::1   1024
rule-xyz        E        ip-vip          NAT     TCP      2001:db8::1   2048-2050
```

ILB 規則の削除

`ilbadm delete-rule` コマンドを使用して、規則を削除します。`-a` オプションを追加して、すべての規則を削除します。次の例では、`rule1` という規則を削除します。

```
# ilbadm delete-rule rule1
```

ユースケース: ILB の構成

このセクションでは、ハーフ NAT トポロジを使用して 2 つのサーバー間でトラフィックの負荷分散を行うように ILB を設定する手順について説明します。52 ページの「ILB の動作モード」の NAT トポロジの実装を参照してください。

1. 管理者になります。

詳細は、『Oracle Solaris 11.2 でのユーザーとプロセスのセキュリティー保護 』の「割り当てられている管理権利の使用」を参照してください。

2. ILB でサーバーグループを設定します。

   2 つのサーバーは `192.168.1.50` と `192.169.1.60` です。次のコマンドを入力して、これら 2 つのサーバーから成るサーバーグループ `srvgrp1` を作成します。ILB のサーバーグループの設定の詳細は、63 ページの「ILB サーバーグループの作成」を参照してください。

   ```
   # ilbadm create-sg -s servers=192.168.1.50,192.168.1.60 srvgrp1
   ```

3. バックエンドサーバーを設定します。

   このシナリオでは、バックエンドサーバーは ILB をデフォルトルーターとして使用するように設定されます。次のコマンドを両方のサーバーで実行します。

   ```
   # route add -p default 192.168.1.21
   ```

   このコマンドを実行したあと、両方のサーバーでサーバーアプリケーションを起動します。ポート 5000 で待機している TCP アプリケーションであるとします。バックエンドサーバーの設定の詳細は、63 ページの「ILB サーバーグループへのバックエンドサーバーの追加」を参照してください。

4. `hc-srvgrp1` という単純な健全性検査を設定します。次のコマンドを入力して、健全性検査を作成します。

   ```
   # ilbadm create-hc -h hc-test=tcp,hc-timeout=3,\
   hc-count=3,hc-interval=60 hc-srvgrp1
   ```

   単純な TCP レベルの健全性検査を使用して、サーバーアプリケーションが到達可能かどうかを検出します。この確認は 60 秒ごとに行われます。健全性検査が最大 3 秒の待機時間をおいて最大 3 回試行し、サーバーが正常かどうかを確認します。試行に 3 回失敗すると、サーバーは `dead` とマークされます。健全性検査のモニタリングおよび作成の詳細は、66 ページの「ILB の健全性検査のモニタリング」を参照してください。

5. 次のコマンドを入力して、ILB 規則を設定します。

   ```
   # ilbadm create-rule -e -p -i vip=10.0.2.20,port=5000 -m \
   lbalg=rr,type=half-nat,pmask=32 \
   -h hc-name=hc-srvgrp1 -o servergroup=srvgrp1 rule1_rr
   ```

   この規則では、持続性 (32 ビットマスク) が使用されます。負荷分散アルゴリズムは `round robin` です。さまざまな ILB アルゴリズムについては、70 ページの「ILB のアルゴリズム」を参照してください。使用されるサーバーグループは `srvgrp1`、使用される健全性検査

メカニズムは hc-srvgrp1 です。ILB 規則の作成の詳細は、70 ページの「ILB 規則の作成」を参照してください。

ILB 統計の表示

このセクションでは、ilbadm コマンドを使用して情報を取得する方法 (サーバーの統計や規則の統計を出力するなど) について説明します。NAT テーブルの情報およびセッション持続性マッピングテーブルを表示することもできます。

統計情報の表示

次の例に示すように、ilbadm show-statistics コマンドを使用して、負荷分散の詳細を表示します。

```
# ilbadm show-statistics
PKT_P   BYTES_P   PKT_U   BYTES_U   PKT_D   BYTES_D
9       636        0        0        0        0
```

PKT_P　処理済みパケット

BYTES_P　処理済みバイト

PKT_U　未処理パケット

BYTES_U　未処理バイト

PKT_D　破棄されたパケット

BYTES_D　破棄されたバイト

NAT 接続テーブルの表示

ilbadm show-nat コマンドを使用して、NAT 接続テーブルを表示します。このコマンドを連続して実行する場合、要素の相対的は位置は重要ではありません。たとえば、ilbadm show-nat 10 コマンドを 2 回実行しても、特にビジー状態のシステムでは、実行ごとに同じ 10 項目が表示

されない場合があります。カウント値を指定しない場合、NAT 接続テーブル全体が表示されます。

例 6-5 NAT 接続テーブルのエントリ

次の例では、NAT 接続テーブルの 5 個のエントリが示されています。

```
# ilbadm show-nat 5
UDP: 124.106.235.150.53688 > 85.0.0.1.1024 >>> 82.0.0.39.4127 > 82.0.0.56.1024
UDP: 71.159.95.31.61528 > 85.0.0.1.1024 >>> 82.0.0.39.4146 > 82.0.0.55.1024
UDP: 9.213.106.54.19787 > 85.0.0.1.1024 >>> 82.0.0.40.4114 > 82.0.0.55.1024
UDP: 118.148.25.17.26676 > 85.0.0.1.1024 >>> 82.0.0.40.4112 > 82.0.0.56.1024
UDP: 69.219.132.153.56132 > 85.0.0.1.1024 >>> 82.0.0.39.4134 > 82.0.0.55.1024
```

このエントリの形式は次のとおりです。

```
T: IP1 > IP2 >>> IP3 > IP4
```

T	このエントリで使用されるトランスポートプロトコル
IP1	クライアントの IP アドレスとポート
IP2	VIP とポート
IP3	ハーフ NAT モードの場合、クライアントの IP アドレスとポート。 フル NAT モードの場合、クライアントの IP アドレスとポート。
IP4	バックエンドサーバーの IP アドレスとポート。

セッション持続性マッピングテーブルの表示

`ilbadm show-persist` コマンドを使用して、セッション持続性マッピングテーブルを表示します。

例 6-6 セッション持続性マッピングテーブルのエントリ

次の例では、セッション持続性マッピングテーブルの 5 個のエントリが表示されます。

```
# ilbadm show-persist 5
rule2: 124.106.235.150 --> 82.0.0.56
rule3: 71.159.95.31 --> 82.0.0.55
rule3: 9.213.106.54 --> 82.0.0.55
rule1: 118.148.25.17 --> 82.0.0.56
rule2: 69.219.132.153 --> 82.0.0.55
```

エントリの形式は次のとおりです。

`R: IP1 --> IP2`

`R`	持続性エントリが関連付けられている規則。
`IP1`	クライアントの IP アドレス。
`IP2`	バックエンドサーバーの IP アドレス。

構成のインポートとエクスポート

エクスポートおよびインポートのサブコマンドは、システム間で構成を移動するために使用されます。たとえば、VRRP を使用してアクティブ/パッシブ構成になるように、ILB のバックアップを設定する場合は、構成をファイルにエクスポートし、バックアップシステムにインポートするだけで済みます。`ilbadm export` コマンドは、現在の ILB 構成をユーザー指定のファイルにエクスポートします。この情報は、あとで `ilbadm import` コマンドへの入力として使用できます。

`ilbadm import` コマンドは、既存の構成を保持するよう特に指示されていないかぎり、インポートの前に既存の構成を削除します。ファイル名が省略されている場合、コマンドは `stdin` から読み取るか、`stdout` に書き込みます。

ILB 構成をエクスポートするには、`export-config` コマンドを使用します。次の例では、`import` コマンドによるインポートに適した形式で、現在の構成をファイル `/var/tmp/ilb_config` にエクスポートします。

```
# ilbadm export-config /var/tmp/ilb_config
```

ILB 構成をインポートするには、`import-config` コマンドを使用します。次の例では、ファイル `/var/tmp/ilb_config` の内容を読み取り、既存の構成をオーバーライドします。

```
# ilbadm import-config /var/tmp/ilb_config
```

♦ ♦ ♦ 第 7 章

7 ILB の高可用性の構成

この章では、VRRP 機能を使用した ILB の高可用性 (HA) の構成について説明します。ILB の高可用性は、DSR およびハーフ NAT トポロジを使用して構成されます。ハーフ NAT および DSR トポロジは、VRRP を使用して、ILB 規則の仮想 IP アドレスを保護します。ただし、ハーフ NAT トポロジでは、VRRP はバックエンドサーバーに面するプライマリロードバランサの IP アドレスの保護にも使用されます。これにより、プライマリロードバランサに障害が発生した場合に、バックエンドサーバーが切り替わり、スタンバイ (パッシブ) ロードバランサを使用することを確認できます。

VRRP の詳細は、第3章「仮想ルーター冗長プロトコルの使用」、ILB の構成方法および管理方法の詳細は、第6章「統合ロードバランサの構成と管理」を参照してください。

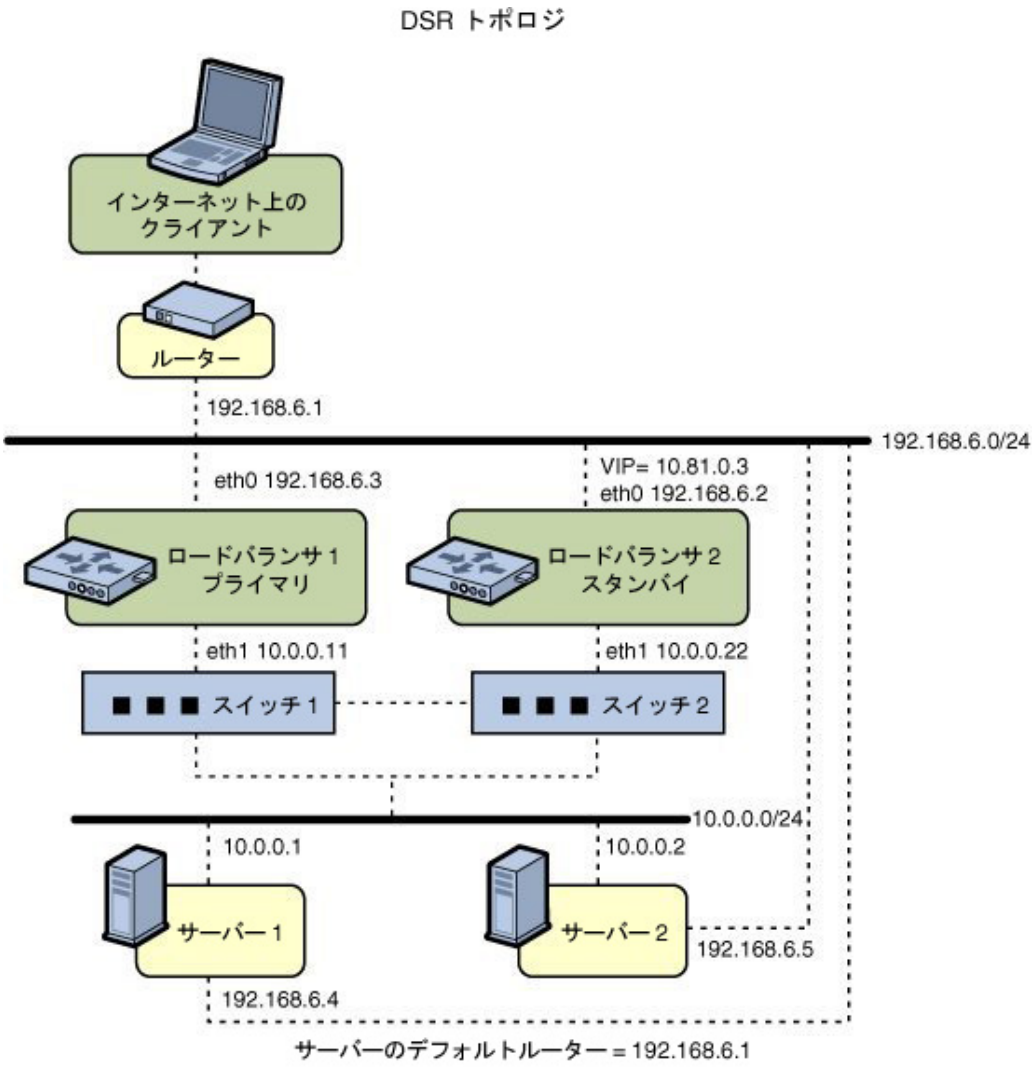
この章の内容は、次のとおりです。

DSR トポロジを使用した ILB の高可用性の構成

2 つのロードバランサの 1 つをプライマリロードバランサ、もう 1 つをスタンバイロードバランサとして設定できます。プライマリロードバランサはマスタールーターとして機能し、スタンバイ (パッシブ) ロードバランサはバックアップルーターとして機能します。ILB 規則の仮想 IP アドレスは、仮想ルーターの IP アドレスとして機能します。VRRP サブシステムは、プライマリロードバランサに障害が発生したかどうかをチェックします。プライマリロードバランサに障害が発生すると、スタンバイロードバランサがプライマリロードバランサの役割を引き受けます。

次の図は、ILB 接続を構成して HA を実現するための DSR トポロジを示しています。

図 7-1 DSR トポロジを使用した ILB の HA 構成



ロードバランサのすべての VIP はサブネット 192.168.6.0/24 に面したインタフェース上に構成されています。

▼ DSR トポロジを使用した ILB の高可用性の構成方法

プライマリロードバランサとスタンバイロードバランサの両方で、ILB 規則、サーバーグループ、および健全性検査の構成が同じになるように構成できます。両方のロードバランサで VRRP を使用するように設定できます。また、規則の仮想 IP アドレスが仮想ルーターアドレスになるように設定します。VRRP サブシステムは、次にロードバランサの 1 つが常にアクティブであることを確認します。

1. **管理者になります。**

   詳細は、『Oracle Solaris 11.2 でのユーザーとプロセスのセキュリティー保護 』の「割り当てられている管理権利の使用」を参照してください。

2. **プライマリロードバランサおよびスタンバイ (パッシブ) ロードバランサの両方の設定が同じになるように構成します。**

```
# ilbadm create-servergroup -s server=10.0.0.1,10.0.0.2 sg1
# ilbadm create-rule -i vip=10.81.0.3,port=9001 \
-m lbalg=hash-ip-port,type=DSR -o servergroup=sg1 rule1
```

3. **ロードバランサ 1 がプライマリロードバランサとして機能するように構成します。**

```
LB1# dladm create-vnic -m vrrp -V 1 -A inet -l eth0 vnic1
LB1# vrrpadm create-router -V 1 -A inet -l eth0 -p 255 vrrp1
LB1# ipadm create-ip vnic1
LB1# ipadm create-addr -d -a 10.81.0.3/24 vnic1
```

   vrrp1 ルーターの優先順位は、vrrpadm コマンドを使用して 255 に設定されます。優先順位の値は、ルーターをマスタールーターにして、アクティブなロードバランサにします。

4. **ロードバランサ 2 がスタンバイロードバランサとして動作するように構成します。**

```
LB2# dladm create-vnic -m vrrp -V 1 -A inet -l eth0 vnic1
LB2# vrrpadm create-router -V 1 -A inet -l eth0 -p 100 vrrp1
LB2# ipadm create-ip vnic1
LB2# ipadm create-addr -d -a 10.81.0.3/24 vnicl
```

   前述の構成は、次の障害シナリオに対する保護を提供します。

   - ロードバランサ 1 に障害が発生すると、ロードバランサ 2 がプライマリロードバランサになります。ロードバランサ 2 は VIP 10.81.0.3 のアドレス解決を引き継ぎ、着信先 IP アドレス 10.81.0.3 を持つクライアントからのすべてのパケットを処理します。

     ロードバランサ 1 が回復すると、ロードバランサ 2 はスタンバイモードに戻ります。

- ロードバランサ 1 の 1 つまたは両方のインタフェースに障害が発生すると、ロードバランサ 2 はプライマリロードバランサとして引き継ぎます。ロードバランサ 2 は VIP `10.81.0.3` のアドレス解決を引き継ぎ、着信先 IP アドレス `10.81.0.3` を持つクライアントからのすべてのパケットを処理します。

  ロードバランサ 1 の両方のインタフェースが正常になると、ロードバランサ 2 はスタンバイモードに戻ります。

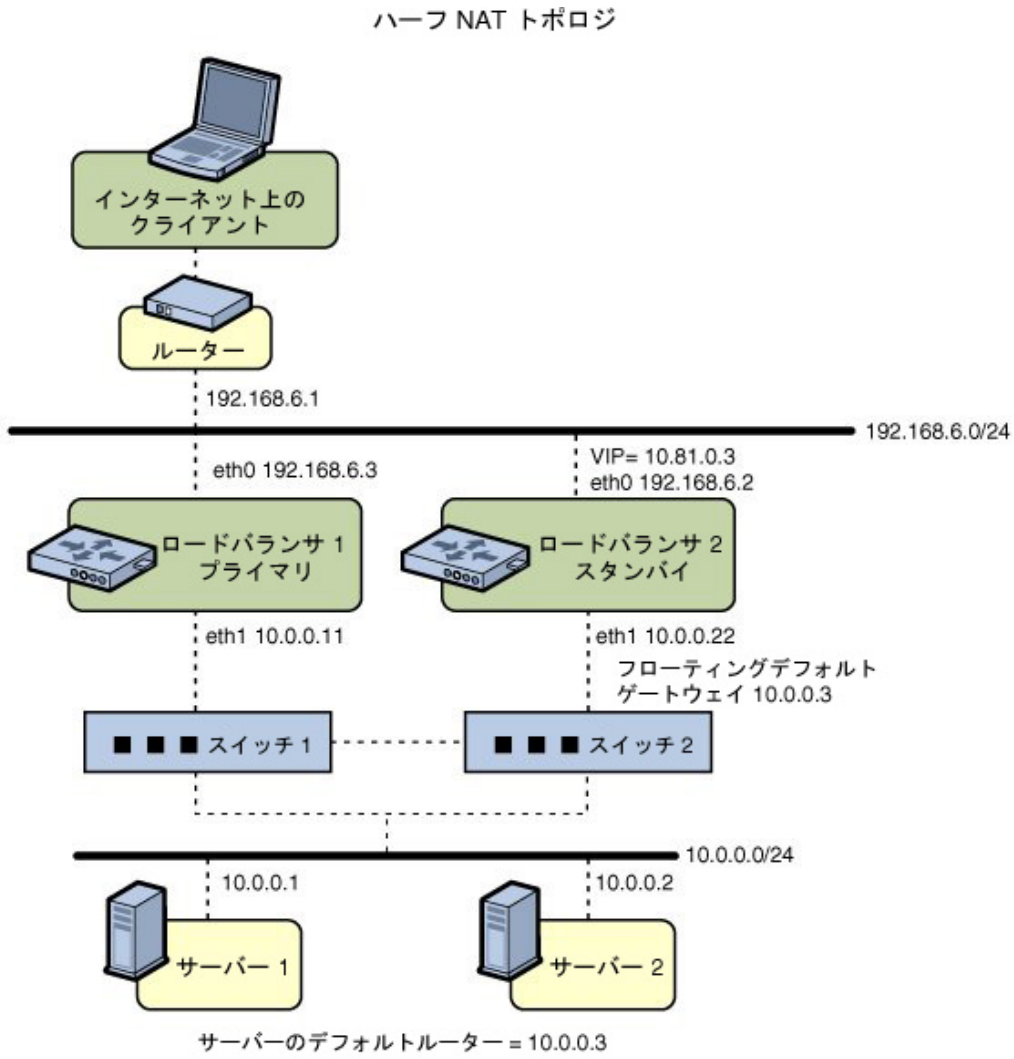
ハーフ NAT トポロジを使用した ILB の高可用性の構成

このセクションでは、ハーフ NAT トポロジを使用することによって、ILB 接続を設定して HA を実現する方法について説明します。2 つのロードバランサを設定する必要があり、1 つはプライマリ、もう 1 つはスタンバイになります。プライマリロードバランサに障害が発生すると、スタンバイロードバランサがプライマリロードバランサの役割を引き受けます。

注記 - ILB の現在の実装では、プライマリロードバランサとスタンバイロードバランサは同期されません。プライマリロードバランサに障害が発生してスタンバイロードバランサが引き継ぐと、既存の接続は失敗します。ただし、同期しない HA でも、プライマリロードバランサに障害が発生した状況では価値があります。

次の図は、ILB 接続を構成して HA を実現するためのハーフ NAT トポロジを示しています。

図 7-2 ハーフ NAT トポロジを使用した ILB の HA 構成



ロードバランサのすべての VIP はサブネット 192.168.6.0/24 に面したインタフェース上に構成されています。

▼ ハーフ NAT トポロジを使用した ILB の高可用性の構成方法

1. **管理者になります。**

   詳細は、『Oracle Solaris 11.2 でのユーザーとプロセスのセキュリティー保護 』の「割り当てられている管理権利の使用」を参照してください。

2. **プライマリロードバランサとスタンバイロードバランサの両方を構成します。**

```
# ilbadm create servergroup -s server=10.0.0.1,10.0.0.2 sg1
# ilbadm create-rule -ep -i vip=10.81.0.3,port=9001-9006,protocol=udp \
-m lbalg=roundrobin,type=HALF-NAT,pmask=24 \
-h hc-name=hc1,hc-port=9006 \
-t conn-drain=70,nat-timeout=70,persist-timeout=70 -o servergroup=sg1 rule1
```

3. **ロードバランサ 1 がプライマリロードバランサとして機能するように構成します。**

```
LB1# dladm create-vnic -m vrrp -V 1 -A inet -l eth0 vnic1
LB1# ipadm create-ip vnic1
LB1# ipadm create-addr -d -a 10.81.0.3/24 vnic1
LB1# vrrpadm create-router -V 1 -A inet -l eth0 -p 255 vrrp1
LB1# dladm create-vnic -m vrrp -V 2 -A inet -l eth1 vnic2
LB1# ipadm create-ip vnic2
LB1# ipadm create-addr -d -a 10.0.0.3/24 vnic2
LB1# vrrpadm create-router -V 2 -A inet -l eth1 -p 255 vrrp2
```

4. **ロードバランサ 2 がスタンバイロードバランサとして動作するように構成します。**

```
LB2# dladm create-vnic -m vrrp -V 1 -A inet -l eth0 vnic1
LB2# ipadm create-ip vnic1
LB2# ipadm create-addr -d -a 10.81.0.3/24 vnic1
LB2# vrrpadm create-router -V 1 -A inet -l eth0 -p 100 vrrp1
LB2# dladm create-vnic -m vrrp -V 2 -A inet -l eth1 vnic2
LB2# ipadm create-ip vnic2
LB2# ipadm create-addr -d -a 10.0.0.3/24 vnic2
LB2# vrrpadm create-router -V 2 -A inet -l eth1 -p 100 vrrp2
```

5. **両方のサーバーにフローティングデフォルトゲートウェイの IP アドレスを追加します。**

```
# route add default 10.0.0.3
```

   この構成は、次の障害シナリオに対する保護を提供します。

   - ロードバランサ 1 に障害が発生すると、ロードバランサ 2 がプライマリロードバランサになります。ロードバランサ 2 は VIP `10.81.0.3` のアドレス解決を引き継ぎ、着信先 IP アドレス `10.81.0.3` を持つクライアントからのすべてのパケットを処理します。ロードバランサ 2 は、フローティングゲートウェイアドレス `10.0.0.3` に送信されるすべてのパケットも処理します。

ロードバランサ 1 が回復すると、ロードバランサ 2 はスタンバイモードに戻ります。

- ロードバランサ 1 の 1 つまたは両方のインタフェースに障害が発生すると、ロードバランサ 2 はプライマリロードバランサとして引き継ぎます。ロードバランサ 2 は VIP `10.81.0.3` のアドレス解決を引き継ぎ、着信先 IP アドレス `10.81.0.3` を持つクライアントからのすべてのパケットを処理します。ロードバランサ 2 は、フローティングゲートウェイアドレス `10.0.0.3` に送信されるすべてのパケットも処理します。

  ロードバランサ 1 の両方のインタフェースが正常になると、ロードバランサ 2 はスタンバイモードに戻ります。

索引

ま

や

ら

B

D

E

G

I

N

O

Q