

在 Oracle® Solaris 11.1 中使用虚拟网络

版权所有 © 2011, 2012, Oracle 和/或其附属公司。保留所有权利。

本软件和相关文档是根据许可证协议提供的，该许可证协议中规定了关于使用和公开本软件和相关文档的各种限制，并受知识产权法的保护。除非在许可证协议中明确许可或适用法律明确授权，否则不得以任何形式、任何方式使用、拷贝、复制、翻译、广播、修改、授权、传播、分发、展示、执行、发布或显示本软件和相关文档的任何部分。除非法律要求实现互操作，否则严禁对本软件进行逆向工程设计、反汇编或反编译。

此文档所含信息可能随时被修改，恕不另行通知，我们不保证该信息没有错误。如果贵方发现任何问题，请书面通知我们。

如果将本软件或相关文档交付给美国政府，或者交付给以美国政府名义获得许可证的任何机构，必须符合以下规定：

U.S. GOVERNMENT END USERS:

Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

本软件或硬件是为了在各种信息管理应用领域内的一般使用而开发的。它不应被应用于任何存在危险或潜在危险的应用领域，也不是为此而开发的，其中包括可能会产生人身伤害的应用领域。如果在危险应用领域内使用本软件或硬件，贵方应负责采取所有适当的防范措施，包括备份、冗余和其它确保安全使用本软件或硬件的措施。对于因在危险应用领域内使用本软件或硬件所造成的一切损失或损害，Oracle Corporation 及其附属公司概不负责。

Oracle 和 Java 是 Oracle 和/或其附属公司的注册商标。其他名称可能是各自所有者的商标。

Intel 和 Intel Xeon 是 Intel Corporation 的商标或注册商标。所有 SPARC 商标均是 SPARC International, Inc 的商标或注册商标，并应按照许可证的规定使用。AMD、Opteron、AMD 徽标以及 AMD Opteron 徽标是 Advanced Micro Devices 的商标或注册商标。UNIX 是 The Open Group 的注册商标。

本软件或硬件以及文档可能提供了访问第三方内容、产品和服务的方式或有关这些内容、产品和服务的信息。对于第三方内容、产品和服务，Oracle Corporation 及其附属公司明确表示不承担任何种类的担保，亦不对其承担任何责任。对于因访问或使用第三方内容、产品或服务所造成的任何损失、成本或损害，Oracle Corporation 及其附属公司概不负责。

目录

前言	5
1 Oracle Solaris 中的网络虚拟化和资源管理	7
网络虚拟化概述	7
网络虚拟化组件	8
谁应实现虚拟网络?	10
用于配置虚拟化组件的命令	10
网络资源管理概述	11
资源控制的数据链路属性	11
通过使用流管理网络资源	12
用于网络资源管理的命令	12
2 在 Oracle Solaris 中创建和管理虚拟网络	15
配置网络虚拟化组件	15
▼ 如何配置 VNIC 和 Etherstub	15
▼ 如何配置带有 VLAN ID 的 VNIC	16
构建虚拟网络	18
▼ 如何为虚拟网络配置区域	18
▼ 如何重新配置区域以使用 VNIC	20
▼ 如何创建专用虚拟网络	22
关于 VNIC 的其他管理任务	25
修改 VNIC 的 VLAN ID	25
修改 VNIC MAC 地址	26
迁移 VNIC	26
显示 VNIC 信息	28
▼ 如何删除 VNIC	28

3	在 Oracle Solaris 中管理网络资源	31
	使用客户机、传送环和接收环	31
	MAC 客户机和环分配	31
	环分配的数据链路属性	32
	使用接收环和传送环的命令	33
	获取和解释环信息	33
	▼如何配置客户机并分配环	37
	使用池和 CPU	39
	▼如何为数据链路配置 CPU 池	40
	▼如何将 CPU 分配给链路	42
	管理流上的资源	43
	▼如何配置流	43
4	监视 Oracle Solaris 中的网络通信和资源使用情况	49
	网络通信流概述	49
	用于监视通信统计数据命令	52
	收集有关链路的网络通信统计数据	52
	获取网络设备的网络通信统计数据	53
	获取通道的网络通信统计数据	54
	获取链路聚合的网络通信统计数据	55
	收集有关流的网络通信统计数据	56
	为网络通信配置网络记帐	57
	▼如何设置网络记账	58
	▼如何获取有关网络通信的历史统计数据	59
	索引	63

前言

欢迎阅读《在 Oracle Solaris 11.1 中使用虚拟网络》。本书是“建立 Oracle Solaris 11.1 网络”系列的一部分，该系列介绍了配置 Oracle Solaris 网络的基本主题和过程。本书假定您已经安装 Oracle Solaris。

目标读者

本书适用于所有负责管理在网络中配置的、运行 Oracle Solaris 的系统的人员。要使用本书，您应当至少具备两年的 UNIX 系统管理经验。参加 UNIX 系统管理培训课程可能会对您有所帮助。

获取 Oracle 支持

Oracle 客户可以通过 My Oracle Support 获取电子支持。有关信息，请访问 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info>，或访问 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>（如果您听力受损）。

印刷约定

下表介绍了本书中的印刷约定。

表 P-1 印刷约定

字体或符号	含义	示例
AaBbCc123	命令、文件和目录的名称；计算机屏幕输出	编辑 .login 文件。 使用 <code>ls -a</code> 列出所有文件。 <code>machine_name% you have mail.</code>
AaBbCc123	用户键入的内容，与计算机屏幕输出的显示不同	<code>machine_name% su</code> Password:
<i>aabbcc123</i>	要使用实名或值替换的命令行占位符	删除文件的命令为 <code>rm filename</code> 。

表 P-1 印刷约定 (续)

字体或符号	含义	示例
<i>AaBbCc123</i>	保留未译的新词或术语以及要强调的词	这些称为 <i>Class</i> 选项。 注意： 有些强调的项目在联机时以粗体显示。
新词术语强调	新词或术语以及要强调的词	高速缓存 是存储在本地的副本。 请勿保存文件。
《书名》	书名	阅读《用户指南》的第 6 章。

命令中的 shell 提示符示例

下表显示了 Oracle Solaris OS 中包含的缺省 UNIX shell 系统提示符和超级用户提示符。请注意，在命令示例中显示的缺省系统提示符可能会有所不同，具体取决于 Oracle Solaris 发行版。

表 P-2 shell 提示符

shell	提示符
Bash shell、Korn shell 和 Bourne shell	\$
Bash shell、Korn shell 和 Bourne shell 超级用户	#
C shell	machine_name%
C shell 超级用户	machine_name#

Oracle Solaris 中的网络虚拟化和资源管理

本章阐述 Oracle Solaris 中关于网络虚拟化和资源管理的基本概念。本章包含以下主题：

- 第 7 页中的“网络虚拟化概述”
- 第 11 页中的“网络资源管理概述”

这些功能可帮助您管理流控制、提高系统性能，并配置所需的网络利用率以便实现 OS 虚拟化、实用程序计算和服务器整合。

网络虚拟化概述

网络虚拟化是将硬件网络资源和软件网络资源组合为单一管理单元的过程。网络虚拟化的目标是为系统和用户提供高效、受控和安全的网络资源共享。

网络虚拟化的最终产品是**虚拟网络**。虚拟网络可分为两大类：外部和内部。**外部虚拟网络**包含作为单个实体由软件进行管理的几个本地网络。传统外部虚拟网络的基本组件是交换机硬件和虚拟局域网 (virtual local area network, VLAN) 软件技术。外部虚拟网络的示例包括大型企业网络和数据中心。

本书侧重于内部虚拟网络。**内部虚拟网络**由一个系统组成，该系统使用的虚拟机或区域的网络接口至少配置在一个物理 NIC 上。这些网络接口称为**虚拟网络接口卡或虚拟 NIC (Virtual Network Interface Card, VNIC)**。这些容器可以相互通信，就像是在同一个本地网络上一样，实际上在一个主机上形成虚拟网络。

一种特殊类型的内部虚拟网络是**专用虚拟网络**。专用虚拟网络和虚拟专用网络 (virtual private networks, VPN) 不同。VPN 软件会在两个端点系统之间创建一个安全的点对点链路。专用虚拟网络是一个系统中不能通过外部系统访问的虚拟网络。该内部网络与其他外部系统的隔离是通过在 etherstub 上配置 VNIC 来实现的。Etherstub 将会在以下章节中介绍。

您可以组合网络资源以配置内部和外部虚拟网络。例如，您可以将使用内部虚拟网络的各个系统配置到 LAN 上，这些 LAN 是一个大型外部虚拟网络一部分。

网络虚拟化组件

以下是 Oracle Solaris 中用于网络虚拟化的基本组件：

- 虚拟网络接口卡 (vNIC)
- 虚拟交换机
- Etherstub

vNIC 是和物理 NIC 有相同数据链路接口的虚拟网络设备。可在底层数据链路上配置 vNIC。当配置 vNIC 后，它们的行为与物理 NIC 一样。此外，系统的资源将 vNIC 视为物理 NIC。vNIC 有自动生成的 MAC 地址。根据使用的网络接口，您可以为 vNIC 显式指定不同于此缺省地址的 MAC 地址，如 `dladm(1M)` 手册页中所述。

有关支持 vNIC 的物理接口的当前列表，请参阅 [Network Virtualization and Resource Control FAQ \(http://hub.opensolaris.org/bin/view/Project+crossbow/faq\)](http://hub.opensolaris.org/bin/view/Project+crossbow/faq)（网络虚拟化和资源控制的常见问题解答）。

创建 vNIC 时，将会自动创建**虚拟交换机**。按照以太网设计，如果一个交换机端口接收从连接到该端口的主机传出的数据包，该数据包无法到达同一端口上的目标。对于使用虚拟网络配置的系统来说，这种设计是一个缺点，因为虚拟网络共享同一个 NIC。传出数据包会穿过交换机端口到达外部网络。因为包不能通过发送它的端口返回，所以传入包不能到达其目标区域。虚拟交换机为这些区域提供了传递包的方法。虚拟交换机为虚拟网络打开了一个相互通信的数据路径。

Etherstub 是伪以太网 NIC。您可以在 *etherstub* 上（而不是在物理链路上）创建 vNIC。*Etherstub* 上的 vNIC 独立于系统上的物理 NIC。使用 *etherstub*，您可以构建同时与系统上的其他虚拟网络和外部网络隔离的专用虚拟网络。例如，如果您要创建一个仅供公司开发人员（而不是整个网络）访问的网络环境，*etherstub* 可以用来创建这样一个环境。

Etherstub 和 vNIC 只是 Oracle Solaris 的虚拟化功能的一部分。通常将这些组件与 Oracle Solaris 区域结合使用。通过指定 vNIC 或 *etherstub* 供区域使用，您可以在单个系统内创建一个网络。有关区域的信息，请参见《[Oracle Solaris 11.1 管理：Oracle Solaris Zones、Oracle Solaris 10 Zones 和资源管理](#)》。

通过组合使用这些组件并将其部署在区域中，您可以得到一个由与下图类似的网络所构成的系统。

图 1-1 单一接口的 VNIC 配置

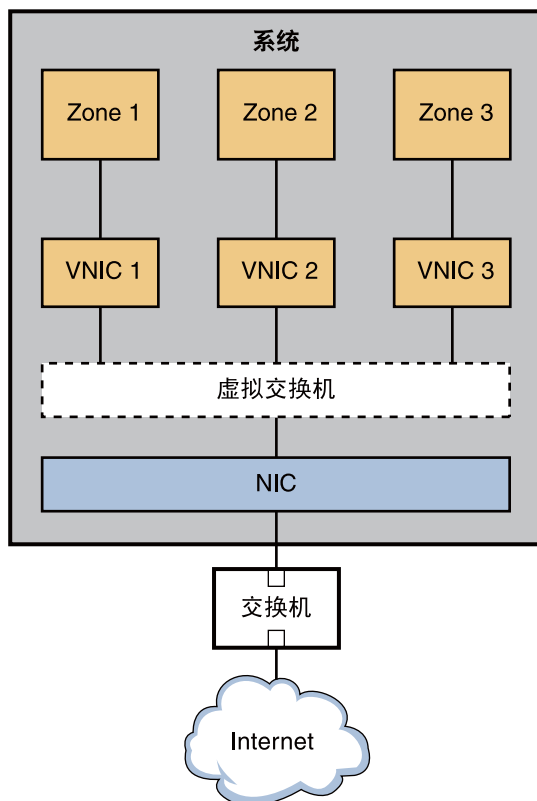


图 1-1 显示了具有一个 NIC 的单个系统。该 NIC 配置了三个 VNIC。每个 VNIC 支持一个区域。区域 1、区域 2 和区域 3 构成该单个系统中的虚拟网络。区域使用各自的 VNIC 相互通信以及与外部网络通信。三个 VNIC 通过虚拟交换机连接到底层物理 NIC。虚拟交换机的作用相当于外部交换机为连接到交换机端口的系统所提供的连接。

配置虚拟网络后，区域将通信发送到外部主机的方式与没有虚拟网络的系统相同。通信从区域经由 VNIC 到达虚拟交换机，然后到达物理接口，由物理接口将数据发送到网络。

还能在系统中的区域之间交换通信。例如，包从区域 1 传递到其专用 VNIC 1。然后，通信经过虚拟交换机到达 VNIC 3。VNIC 3 再将通信传递到区域 3。该通信从未离开过系统，因此永远不会违反以太网限制。

您也可以创建一个基于 etherstub 的虚拟网络。Etherstub 完全基于软件，不需要有网络接口作为虚拟网络的基础。

谁应实现虚拟网络？

如果您需要整合 Oracle 的 Sun 服务器上的资源，可以考虑实施 VNIC 和虚拟网络。ISP、电讯公司和大型金融机构的整合人员可以使用以下网络虚拟化功能提高其服务器和网络性能。

- NIC 硬件，包括支持硬件环的功能强大的新接口
- 各 VNIC 的多个 MAC 地址
- 新接口提供的大量带宽

您可以用包含多个区域或虚拟机的单个系统取代多个系统，而不会明显损失隔离性、安全性和灵活性。

有关网络虚拟化优点的演示，请参见 [Consolidating the Data Center With Network Virtualization \(http://download.oracle.com/otndocs/tech/OTN_Demos/data-center-consolidation.html\)](http://download.oracle.com/otndocs/tech/OTN_Demos/data-center-consolidation.html)（使用网络虚拟化合并数据中心）。

用于配置虚拟化组件的命令

要创建 VNIC，请使用 `dladm create-vnic` 命令。

```
# dladm create-vnic -l link [-v vid] vnic
```

`-l link` 指在其上配置 VNIC 的数据链路的名称。

`-v vid` 指 VNIC 的 VLAN ID（如果要为 VNIC 创建 VLAN）。此选项不是必需选项。要配置具有 VLAN ID 的 VNIC，请参见第 16 页中的“如何配置带有 VLAN ID 的 VNIC”。有关 VLAN 的更多信息，请参见《管理 Oracle Solaris 11.1 网络性能》中的第 3 章“使用 VLAN”。

`vnic` 指 VNIC 的名称。

注 - 您可以为 VNIC 配置其他属性，例如 MAC 地址、与 VNIC 关联的 CPU 等等。有关包含这些属性的列表，请参阅 `dladm(1M)` 手册页。某些属性修改只适用于 VNIC。例如，通过 `dladm create-vnic` 命令，您可以配置一个 MAC 地址并指定 VLAN ID 以将 VNIC 创建为 VLAN。不过，您不能使用 `dladm create-vlan` 命令直接为 VLAN 配置 MAC 地址。

一次只能在一个数据链路上创建一个 VNIC。作为数据链路，VNIC 具有您可以按需进一步配置的链路属性。第 11 页中的“资源控制的数据链路属性”列出了部分用于管理系统中的网络资源使用的属性。

要创建 `etherstub`，请使用 `dladm create-ether` 命令。

```
# dladm create-ether etherstub
```

创建 VNIC 或 etherstub 只是配置虚拟网络的预备步骤。要使用这些组件在系统上创建虚拟网络，请参见第 2 章，在 [Oracle Solaris 中创建和管理虚拟网络](#)。

网络资源管理概述

本节阐述了可用来管理系统上网络资源使用的不同方法。

资源控制的数据链路属性

在 Oracle Solaris 11 中，可以通过管理网络资源来更轻松且动态地实现服务质量 (quality of service, QoS)。网络资源管理包括设置与网络资源相关的数据链路属性。通过设置这些属性，您可以确定给定资源在多大程度上可用于网络处理。例如，一个链路可以与特定数量的专门为网络处理保留的 CPU 相关联。也可以为一个链路指定给定带宽来处理特定类型的网络通信。

定义资源属性后，新值将立即生效。此方法使资源管理更加灵活。您可以在创建链路时设置资源属性。也可以稍后设置这些属性，例如：在了解资源使用情况一段时间并确定如何更好地分配资源之后再设置这些属性。资源分配过程既适用于虚拟网络环境，也适用于传统的物理网络。例如，使用 `dladm set-linkprop` 命令可设置与网络资源相关的属性。物理和虚拟数据链路使用相同的语法。

网络资源管理相当于为通信创建专用通道。当您组合不同的资源以满足特定类型的网络数据包时，这些资源就形成一个用于这些数据包的**网络通道**。可以为每个网络通道以不同的方式指定资源。例如，您可以为网络通信流量最大的通道分配更多资源。通过配置根据实际需要分配资源的网络通道，可以提高系统处理数据包的效率。有关网络通道的更多信息，请参见第 49 页中的“[网络通信流概述](#)”。

网络资源管理有助于完成以下任务：

- 网络筹备
- 建立服务级别协议
- 客户机记帐
- 诊断安全问题

您可以对单个系统上的数据通信进行隔离、设置优先级、跟踪和控制，而无需使用复杂的 QoS 规则定义。

通过使用流管理网络资源

流是为了进一步控制如何使用资源来处理数据包而对这些数据包进行分类的定制方式。网络数据包可以根据属性分类。共享同一属性的数据包构成一个流并标有特定的流名称。然后可以为流指定特定的资源。

用作创建流的基础的属性是从包头信息派生的。您可以根据以下属性之一将包通信划分为流：

- IP 地址
- 传输协议名称（UDP、TCP 或 SCTP）
- 应用程序端口号（例如，用于 FTP 的端口 21）
- DS 字段属性，仅用于 IPv6 包中的 QoS。有关 DS 字段的更多信息，请参阅《在 Oracle Solaris 11.1 中管理 IP 服务质量》。

流只能基于列表中的属性之一。例如，您可以根据正在使用的端口（如用于 FTP 的端口 21）或者根据 IP 地址（如来自特定源 IP 地址的数据包）创建一个流。但是，您无法为在端口 21 上接收到的来自指定 IP 地址的包创建一个流。同样，您不能为来自 IP 地址 192.168.1.10 的所有通信创建一个流，然后为 192.168.1.10 上的传输层通信创建一个流。因此，您可以在一个系统上配置多个流，其中每个流基于不同的属性。

用于网络资源管理的命令

用来分配网络资源的命令会根据您是直接操作数据链路还是操作流而有所不同。

- 对于数据链路，要根据是在创建链路的同时设置属性还是设置现有链路的属性，使用适当的 `dladm` 子命令。要创建一个链路并同时将其资源分配给它，请使用以下语法：

```
# dladm create-vnic -l link -p property=value[,property=value] vnic
```

其中 `link` 可以是物理链路或虚拟链路。

要设置现有链路的属性，请使用以下语法：

```
# dladm set-linkprop -p property=value[,property=value] link
```

以下是您可以为资源分配设置的链路属性：

- **带宽**—您可以限制供特定链路使用的硬件带宽。
- **NIC 环**—如果 NIC 支持环分配，可将其传送和接收环指定为专供数据链路使用。NIC 环将在第 31 页中的“使用客户机、传送环和接收环”中介绍。
- **CPU 池**—CPU 池通常是使用特定区域创建的并与之相关联。这些池可以指定给数据链路以保留 CPU 集，以便管理其相关联的区域的网络处理。CPU 和池将在第 39 页中的“使用池和 CPU”中介绍。

- **CPU**—在具有多个 CPU 的系统上，您可以分配给定数量的 CPU 专用于特定网络处理。
- 对于流，您可以使用 `flowadm` 子命令。管理流上资源的方法与管理数据链路上资源的方法类似。要创建一个流并向其添加资源，请使用以下语法：

```
# flowadm add-flow -l link -a attribute=value[,attribute=value] \  
-p property=value[,property=value] flow
```

表明流特征的已定义属性集构成系统的**流控制策略**。

要设置现有流的属性，请使用以下语法：

```
# flowadm set-flowprop -p property=value[,property=value] flow
```

可以指定给流的资源分配属性与直接指定给链路的属性相同。不过，目前只有带宽属性可以与流相关联。虽然用于设置属性的命令对于数据链路和流是不同的，但语法类似。要配置带宽属性，请参见第 43 页中的“如何配置流”中的示例

有关 `flowadm` 命令的更多详细信息，请参阅 `flowadm(1M)` 手册页。要获取与 `flowadm` 命令一起使用的子命令的列表，请键入以下内容：

```
# flowadm help  
The following subcommands are supported:  
Flow      : add-flow      remove-flow    reset-flowprop  
           set-flowprop  show-flow     show-flowprop  
For more info, run: flowadm help <subcommand>.
```


在 Oracle Solaris 中创建和管理虚拟网络

本章包含有关在单个系统中配置虚拟网络的任务。

本章包含以下主题：

- 第 15 页中的“配置网络虚拟化组件”
- 第 18 页中的“构建虚拟网络”
- 第 25 页中的“关于 VNIC 的其他管理任务”

有关虚拟网络的介绍，请参见第 1 章，Oracle Solaris 中的网络虚拟化和资源管理。

配置网络虚拟化组件

在 Oracle Solaris 11 中，etherstub 和 VNIC 是网络虚拟化的基本组件。本节介绍了用于配置这些组件以便构建虚拟网络的步骤。有关这些组件的说明，请参见第 8 页中的“网络虚拟化组件”。

本节介绍了以下过程：

- 第 15 页中的“如何配置 VNIC 和 Etherstub”
- 第 16 页中的“如何配置带有 VLAN ID 的 VNIC”

▼ 如何配置 VNIC 和 Etherstub

VNIC 将虚拟网络连接到外部网络。VNIC 还能够使区域通过 VNIC 自动创建的虚拟交换机相互通信。对于托管区域间的内部通信以及区域与外部 LAN 和 Internet 之间的通信的虚拟网络，每个区域都必须具有自己的接口。因此，必须根据虚拟网络中要配置的区域的数量来重复此过程。

1 成为管理员。

有关更多信息，请参见《Oracle Solaris 11.1 管理：安全服务》中的“如何使用指定给您的管理权限”。

2 (可选) 创建 etherstub。

```
# dladm create-etherstub etherstub
```

仅当要创建限制外部系统访问的专用虚拟网络时，才应执行此步骤。有关专用虚拟网络的说明，请参见第 7 页中的“网络虚拟化概述”。

与数据链路一样，您可以采用任何对您的网络设置有意义的方式来命名 etherstub。有关创建定制名称的准则，请参见《Oracle Solaris 11 联网介绍》中的“有效链路名称的规则”。

3 创建 VNIC。

```
# dladm create-vnic -l datalink [-v vid] vnic
```

如果要为专用虚拟网络创建 VNIC，请为 datalink 指定一个 etherstub。仅当将 VNIC 创建为 VLAN 时，才应在命令语法中包含 -v vid，其中，vid 指的是 VNIC 的 VLAN ID。否则，省略此选项。

如果要将 VNIC 创建为 VLAN，请参阅第 16 页中的“如何配置带有 VLAN ID 的 VNIC”以了解特定于作为 VLAN 的 VNIC 的附加步骤。

您可以为 VNIC 指定任何名称。要为 VNIC 指定定制名称，请参见《Oracle Solaris 11 联网介绍》中的“有效链路名称的规则”。

4 创建 VNIC 上的 IP 接口。

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

5 为 VNIC 接口指定一个静态 IP 地址。

```
# ipadm create-addr -a address interface
```

-a address 指定 IP 地址，该地址可以采用 CIDR 表示法。

interface 指定上一步中创建的 VNIC。

静态 IP 地址可以为 IPv4 或 IPv6 地址。有关配置 IP 地址的更多信息，请参见《在 Oracle Solaris 11.1 中使用固定网络配置连接系统》中的“如何配置 IP 接口”。

有关配置 IP 地址的更多信息，请参见《在 Oracle Solaris 11.1 中使用固定网络配置连接系统》中的“如何配置 IP 接口”。

6 将地址信息添加到 /etc/hosts 文件中。

▼ 如何配置带有 VLAN ID 的 VNIC

在虚拟网络中，您可以配置带有 VLAN ID 的 VNIC 以托管 VLAN 通信。您还可以设置链路属性 vlan-announce 以将每个 VNIC 的 VLAN 配置传播到网络。

与常规 VLAN 链路不同，配置为 VLAN 的 VNIC 有自己的 MAC 地址。有关非 VNIC VLAN 的信息，请参见《管理 Oracle Solaris 11.1 网络性能》中的第 3 章“使用 VLAN”。

注 – 以下过程仅包含用于创建带有 VLAN ID 的 VNIC 的步骤以及用于设置可使 VNIC 提供 VLAN 通信服务的相应属性的步骤。尽管在启用该属性时会自动更新中间端口和交换机，但是，必须分别配置端点以便在这些端点上定义 VLAN。

1 成为管理员。

有关更多信息，请参见《Oracle Solaris 11.1 管理：安全服务》中的“如何使用指定给您的管理权限”。

2 创建带有 VLAN ID 的 VNIC。

```
# dladm create-vnic -l link -v vid vnic
```

3 将 VNIC 的 VLAN 配置广播到网络。

```
# dladm set-linkprop -p vlan-announce=gvrp link
```

通过此步骤可启用 GARP VLAN 注册协议 (GARP VLAN Registration Protocol, GVRP) 客户机系统，该系统自动在连接的交换机中注册 VLAN ID。缺省情况下，`vlan-announce` 属性设置为 `off`，表明不会将任何 VLAN 广播消息发送到网络。将该属性设置为 `gvrp` 后，会传播该链路的 VLAN 配置以启用网络设备的自动 VLAN 端口配置。因此，这些设备可以接受 VLAN 通信并将其转发。

4 (可选) 要配置 VLAN 广播之间的等待时间，请设置 `gvrp-timeout` 属性。

```
# dladm set-linkprop -p gvrp-timeout=time link
```

其中，`time` 以毫秒为单位。缺省值为 250 毫秒。负载很重的系统在重新广播 VLAN 信息时需要一个较短的时间间隔。通过该属性可调整时间间隔。

5 (可选) 要显示属性 `vlan-announce` 和 `gvrp-timeout` 的值，请使用以下命令：

```
# dladm show-linkprop -p vlan-announce,gvrp-timeout
```

示例 2-1 将 VNIC 配置为 VLAN

本示例创建带有 VLAN ID 的 VNIC，并启用要广播到网络的 VLAN 配置。

```
# dladm create-vnic -l net0 -v 123 vnic0
# dladm set-linkprop -p vlan-announce=gvrp net0
# dladm show-linkprop -p vlan-announce,gvrp-timeout net0
LINK    PROPERTY      PERM  VALUE  DEFAULT  POSSIBLE
net0    vlan-announce rw    gvrp   off      gvrp,off
net0    gvrp-timeout  rw    250    250     --
```

构建虚拟网络

虚拟网络组合使用了区域和虚拟化组件。可根据系统可支持的程度创建尽可能多的区域。每个区域都有自己的虚拟接口。系统中的区域可相互通信。虚拟网络作为一个整体连接到大型外部网络上的目标。

构建虚拟网络包括用于配置 etherstub 或 VNIC 的一个或多个步骤以及用于配置区域的步骤。尽管这些是独立的过程集，但是必须同时执行才能完成虚拟网络的构建。

本节中的过程基于以下假设：

- 系统上的虚拟网络包含三个区域。这些区域处于不同的配置阶段：第一个区域被创建为新区域，第二个区域已存在于系统上并需要重新配置才能使用 VNIC，第三个区域被指定为专用虚拟网络。因此，这些过程说明了用于为虚拟网络准备区域的各种方式。
- 系统物理接口的 IP 地址配置为 192.168.3.70
- 路由器的 IP 地址为 192.168.3.25

在本节的每个过程中，方案中添加了更多的详细信息以便为步骤提供更具体的上下文。

构建虚拟网络时，一些步骤需在全局区域中执行，一些步骤需在非全局区域中执行。为清楚起见，每个过程后面的示例中的提示将指明在哪个区域中发出特定命令。但是，提示显示的实际路径可能有所不同，具体取决于特定于您系统的提示。

在本节中，讨论了以下过程：

- [第 18 页](#)中的“[如何为虚拟网络配置区域](#)”
- [第 20 页](#)中的“[如何重新配置区域以使用 VNIC](#)”
- [第 22 页](#)中的“[如何创建专用虚拟网络](#)”

▼ 如何为虚拟网络配置区域

此过程介绍了如何使用新 VNIC 配置新区域。请注意，此过程仅包含与网络虚拟化相关的步骤。有关配置区域的更多详细说明，请参阅《[Oracle Solaris 11.1 管理：Oracle Solaris Zones、Oracle Solaris 10 Zones 和资源管理](#)》中的第 17 章“[规划和配置非全局区域（任务）](#)”。

此过程假设将虚拟网络的第一个区域创建为一个全新的区域。

1 成为管理员。

有关更多信息，请参见《[Oracle Solaris 11.1 管理：安全服务](#)》中的“[如何使用指定给您的管理权限](#)”。

2 配置 VNIC。

请参见第 15 页中的“如何配置 VNIC 和 Etherstub”。但是，对于此特定过程，请省略创建 etherstub 的步骤。

3 创建区域。

```
global# zonecfg -z zone
```

创建区域时，请确保将 ip-type 参数设置为 exclusive，并确保将刚刚创建的 VNIC 指定为区域的物理接口。

4 要退出区域配置模式，请验证之后再提交该配置。**5 安装区域。**

```
global# zoneadm -z zone install
```

注 - 安装过程可能需要一些时间。

6 启动区域。

```
global# zoneadm -z zone boot
```

7 在区域完全启动后，登录到该区域。

```
# zlogin -C zone
```

8 根据提示提供信息。

大部分信息通过从选项列表中进行选择来提供。通常，缺省选项即可满足需求。要配置虚拟网络，必须提供或验证以下信息：

- 区域的主机名，例如 zone1。
- 区域的 IP 地址（它基于区域的 VNIC 的 IP 地址）。
- 是否应启用 IPv6。
- 具有虚拟网络的系统是否为子网的一部分。
- IP 地址的网络掩码。
- 缺省路由，可以是在其上构建虚拟网络的物理接口的 IP 地址。

提供所需的信息后，该区域将重新启动。

示例 2-2 为虚拟网络配置区域

本示例包含用于创建 zone1 的详细步骤。但是，只列出了与创建虚拟网络相关的区域参数。

```
global # zonecfg -z zone1
zonecfg:zone1> create
zonecfg:zone1> set zonepath=/export/home/zone1
```

```
zonecfg:zone1> set autoboot=true
zonecfg:zone1> set ip-type=exclusive
zonecfg:zone1> add net
zonecfg:zone1:net> set physical=vnic1
zonecfg:zone1:net> end
zonecfg:zone1> verify
zonecfg:zone1> commit
zonecfg:zone1> exit

global# zoneadm -z zone1 install
Preparing to install zone <zone1>
Creating list of files to copy from the global zone.
.
.
Zone <zone1> is initialized.

global# zoneadm -z zone1 boot

zlogin -C zone1
What type of terminal are you using?
.
.
.
8) Sun Workstation
9) Televideo 910
10) Televideo 925
11) Wyse Model 50
12) X Terminal Emulator (xterms)
13) CDE Terminal Emulator (dtterm)
14) Other
Type the number of your choice and press Return: 13
.
(More prompts)
..
```

对于网络信息，提供以下信息：

```
Hostname: zone1
IP address: 192.168.3.80
System part of a subnet: Yes
Netmask: 255.255.255.0
Enable IPv6: No
Default route: 192.168.3.70
Router IP address: 192.168.3.25
```

▼ 如何重新配置区域以使用 VNIC

此过程中所提及的区域指的是虚拟网络中的第二个区域。该区域已存在，但其当前配置会阻止它成为虚拟网络的一部分。具体地说，该区域的 IP 类型是共享类型，而其当前接口是 net0。这两个配置必须更改。

- 1 成为管理员。

有关更多信息，请参见《Oracle Solaris 11.1 管理：安全服务》中的“如何使用指定给您的管理权限”。

- 2 创建 VNIC。

```
global# dladm create-vnic [-v vid] -l datalink vnic
```

其中，*vid* 指的是为 VNIC 指定的 VLAN ID。仅当将 VNIC 创建为 VLAN 时，才应指定 VLAN ID。

现在请勿配置 VNIC 的接口。将稍后执行此步骤。

- 3 将区域的 IP 类型从共享更改为专用。

```
global# zonecfg -z zone
zonecfg:zone1> set ip-type=exclusive
zonecfg:zone1>
```

- 4 更改区域的接口以使用 VNIC。

```
zonecfg:zone1> remove net physical=NIC
zonecfg:zone1> add net
zonecfg:zone1:net> set physical=vnic
zonecfg:zone1:net> end
zonecfg:zone1>
```

- 5 验证并提交您已经实施的更改，然后退出该区域。

```
zonecfg:zone1 verify
zonecfg:zone1> commit
zonecfg:zone1> exit
global#
```

- 6 重新引导区域。

```
global# zoneadm -z zone reboot
```

- 7 登录到区域。

```
global# zlogin zone
```

- 8 使用有效的 IP 地址配置 VNIC。

如果您要为 VNIC 指定一个静态地址，需要键入以下内容：

```
zone# ipadm create-addr -a address interface
```

其中，*address* 可以使用 CIDR 表示法。

- 9 在全局区域中，将地址信息添加到 `/etc/hosts` 文件中。

示例 2-3 重新配置区域配置以使用 VNIC

在本示例中，zone2 已作为共享区域存在。此外，该区域使用的是系统的主接口，而不是虚拟链路。您需要修改 zone2 才能使用 vnic2。要使用 vnic2，必须首先将 zone2 的 IP 类型更改为专用。请注意，某些输出已截断，以重点显示与虚拟网络相关的信息。

```
global# dladm create-vnic -l net0 vnic2

global# zonecfg -z zone2
zonecfg:zone1> set ip-type=exclusive
zonecfg:zone1> remove net physical=net0
zonecfg:zone1> add net
zonecfg:zone1:net> set physical=vnic2
zonecfg:zone1:net> end
zonecfg:zone1> verify
zonecfg:zone1> commit
zonecfg:zone1> exit
global# zoneadm -z zone2 reboot

global# zlogin zone2
zone2# ipadm create-ip vnic2
zone2# ipadm create-addr -a 192.168.3.85/24 vnic2
ipadm: vnic2/v4

zone2# exit

global# vi /etc/hosts
#
::1          localhost
127.0.0.1    localhost
192.168.3.70 loghost    #For net0
192.168.3.80 zone1     #using vnic1
192.168.3.85 zone2     #using vnic2
```

▼ 如何创建专用虚拟网络

以下过程介绍了如何配置虚拟网络的第三个区域。尽管该区域属于虚拟网络，但外部系统无法对其进行访问。要使隔离区域发送跨越系统的网络通信，则必须使用网络地址转换(network address translation, NAT)。NAT 将 VNIC 的专用 IP 地址转换为物理网络接口的可路由 IP 地址。但是，专用 IP 地址本身对外部网络不可见。有关 NAT 的更多信息，请参见《在 Oracle Solaris 11.1 中保护网络安全》中的“使用 IP 过滤器的 NAT 功能”。

是否使用 etherstub 是常规虚拟网络与专用虚拟网络之间的主要区别。在专用虚拟网络中，指定给区域的 VNIC 是在 etherstub 上配置的。因此，它们与流经系统的网络通信相互隔离。

以下过程假设该区域已存在，但当前不包含任何关联接口。

1 成为管理员。

有关更多信息，请参见《Oracle Solaris 11.1 管理：安全服务》中的“如何使用指定给您的管理权限”。

2 创建 etherstub。

```
global# dladm create-etherstub etherstub
```

3 在 etherstub 上创建一个 VNIC。

```
global# dladm create-vnic -l etherstub vnic
```

现在请勿配置 VNIC 的接口。将稍后执行此步骤。

4 将 VNIC 指定给该区域。

```
global# zonecfg -z zone
zone# set physical=vnic
```

5 验证并提交您已经实施的更改，然后退出该区域。

```
zonecfg:zone1 verify
zonecfg:zone1> commit
zonecfg:zone1> exit
global#
```

6 登录到区域。

```
# zlogin zone
```

7 在该区域中，通过现已指定给该区域的 VNIC 创建 IP 接口。

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

8 使用有效的 IP 地址配置 VNIC。

如果您要为 VNIC 指定一个静态地址，需要键入以下内容：

```
zone# ipadm create-addr -a address interface
```

其中，*address* 可以使用 CIDR 表示法。

9 在全局区域中，将地址信息添加到 /etc/hosts 文件中。**10 在全局区域中，设置主接口以执行 IP 转发。**

```
# ipadm set-ifprop -p forwarding=on -m ipv4 primary-interface
```

注 – 在 Oracle Solaris 11 中，主接口通常使用名称 net0。

11 在全局区域中，在 /etc/ipnat.conf 文件中为主接口配置网络地址转换 (network address translation, NAT)。

12 启动 IP 过滤服务以启用 NAT。

```
# svcadm enable network/ipfilter
```

13 重新引导区域。

```
# zoneadm -z zone reboot
```

示例 2-4 创建专用虚拟网络配置

在本示例中，将 zone3 配置为要隔离的专用网络。同时，将 NAT 和 IP 转发配置为允许虚拟专用网络向主机外部发送包，但仍对外部网络隐藏其专用地址。已使用专用 IP 类型配置了该区域。但是，未为其指定任何 IP 接口。

```
global# dladm create-etherstub ether0
global# dladm create-vnic -l ether0 vnic3
global# zonecfg -z zone3
zonecfg:zone3> add net
zonecfg:zone3:net> set physical=vnic3
zonecfg:zone3:net> end
zonecfg:zone3> verify
zonecfg:zone3> commit
zonecfg:zone3> exit
global#

global# zlogin zone3
zone3# ipadm create-ip vnic3
zone3# ipadm create-addr -a 192.168.0.10/24 vnic3
ipadm: vnic3/v4
zone3# exit

global# cat /etc/hosts
::1          localhost
127.0.0.1    localhost
192.168.3.70 loghost    #For net0
192.168.3.80 zone1      #using vnic1
192.168.3.85 zone2      #using vnic2
192.168.0.10 zone3      #using vnic3

global# ipadm set-ifprop -p forwarding=on -m ipv4 vnic3

global# vi /etc/ipf/ipnat.conf
map vnic3 192.168.0.0/24 -> 0/32 portmap tcp/udp auto
map vnic3 192.168.0.0/24 -> 0/32

global# svcadm enable network/ipfilter
global# zoneadm -z zone3 boot
```


关于 VNIC 的其他管理任务

本节介绍了在执行完基本配置之后可对 VNIC 执行的任务。本节包含以下主题：

- 第 25 页中的“修改 VNIC 的 VLAN ID”
- 第 26 页中的“修改 VNIC MAC 地址”
- 第 26 页中的“迁移 VNIC”
- 第 28 页中的“显示 VNIC 信息”
- 第 28 页中的“如何删除 VNIC”

注 - VNIC 可配置为 VLAN。通过一个类似的子命令 `dladm modify-vlan` 可以修改已使用 `dladm create-vlan` 命令创建的直接 VLAN。必须根据要修改 VLAN 还是修改配置为 VLAN 的 VNIC 来使用正确的子命令。对 `dladm show-vlan` 子命令所显示的 VLAN 使用 `modify-vlan` 子命令。对 `dladm show-vnic` 子命令所显示的 VNIC（包括带有 VLAN ID 的 VNIC）使用 `modify-vnic` 子命令。要修改直接 VLAN，请参见《[管理 Oracle Solaris 11.1 网络性能](#)》中的“修改 VLAN”。

以下是两种可用的 VNIC 修改方式：

- 全局修改，将同时更改特定数据链路上所有 VNIC 的某个属性。使用 `-L` 选项标识要修改其 VNIC 的底层数据链路。
- 选择性修改，将修改所选 VNIC 的某个属性。指定要更改其属性的 VNIC，而不是使用 `-L` 选项标识底层数据链路。

您可以修改以下属性：VLAN ID、MAC 地址和底层链路。修改底层链路意味着将 VNIC 移动到另一个数据链路。以下各节详细讨论了这些修改。

修改 VNIC 的 VLAN ID

要更改 VNIC 的 VLAN ID，请使用以下命令之一：

- `dladm modify-vnic -v vid -L datalink`

在此命令中，`vid` 指定为 VNIC 分配的新 VLAN ID。`datalink` 指的是在其上配置 VNIC 的底层链路。如果该数据链路中仅存在一个 VNIC，可使用此命令语法。无法对已配置有多个 VNIC 的数据链路执行此命令，因为这些 VNIC 都必定具有唯一的 VLAN ID。

- `dladm modify-vnic -v vid vnic`

使用此命令可更改一个数据链路上的多个 VNIC 的唯一 VLAN ID。因为每个 VLAN ID 对于同一数据链路上的 VNIC 都是唯一的，所以，一次只能更改一个 VLAN ID。假设要更改在 `net0` 上配置的 `vnic0`、`vnicb0` 和 `vnicc0` 的 VLAN ID。应按照以下方式执行操作：

```
# dladm modify-vnic -v 123 vnic0
# dladm modify-vnic -v 456 vnicb0
# dladm modify-vnic -v 789 vnicc0
```

- `dladm modify-vnic -v vid vnic, vnic, [...]`

使用此命令可更改作为一个组的多个 VNIC 的 VLAN ID，假设每个 VNIC 位于不同的数据链路上。假设要更改 `vnic0`、`vnic1` 和 `vnic2` 的 VLAN ID。这些 VNIC 分别配置在 `net0`、`net1` 和 `net2` 上。可使用以下命令：

```
# dladm modify-vnic -v 123 vnic0,vnic1,vnic2
```

修改 VNIC MAC 地址

VNIC 具有唯一的 MAC 地址。要修改这些地址，请使用以下任一适用于您特定环境的命令：

- `dladm modif-vnic -m mac-address vnic`

使用此命令可为特定 VNIC 指定特定 MAC 地址。

- `dladm modif-vnic -m random -L datalink`

此命令执行的是全局修改，可更改数据链路上所有 VNIC 的 MAC 地址。系统自动为 VNIC 指定唯一的 MAC 地址。在此命令中，`-m random` 选项等效于 `-m auto` 选项。

- `dladm modif-vnic -m random vnic, vnic, [...]`

此命令执行的是选择性的 VNIC 修改。请注意，对于全局修改和选择性修改，`-m` 选项均需指定为 `random`。

您可以使用单个命令修改 VNIC 的 VLAN ID 和 MAC 地址。但是，使用命令全局修改多个 VNIC 属性时，务必要小心，因为这可能会导致意外行为。一次更改一个 VNIC 的多个属性要比同时更改一组中所有 VNIC 的多个属性可取。

以下示例显示了在修改 VNIC 的 VLAN ID 和 MAC 地址前后的输出：

```
# dladm show-vnic vnic0
LINK    OVER    SPEED  MACADDRESS      MACADDRTYPE  VID
vnic0   net0    1000   2:8:20:ec:c4:1d random         0
# dladm modify-vnic -m random -v 123 vnic0
# dladm show-vnic vnic0
LINK    OVER    SPEED  MACADDRESS      MACADDRTYPE  VID
vnic0   net0    1000   2:8:20:0:1:2    random        123
```

迁移 VNIC

您可以将一个或多个 VNIC 从一个底层数据链路移动到另一个底层数据链路，而无需删除和重新配置 VNIC。底层链路可以是物理链路、链路聚合或 `etherstub`。

要成功迁移 VNIC，向其移动 VNIC 的底层数据链路必须能够适用该 VNIC 的数据链路属性。如果不支持这些属性，则迁移失败，系统将通知用户。成功迁移后，在这些 VNIC 仍与网络保持连接的情况下，使用这些 VNIC 的所有应用程序将继续正常运行。

某些与硬件相关的属性在 VNIC 迁移后可能会发生更改，例如数据链路状态、链路速度、MTU 大小等。这些属性的值继承自向其迁移 VNIC 的数据链路。

您还可以全局或选择性地迁移 VNIC。全局迁移即将一个数据链路上的所有 VNIC 均迁移到另一个数据链路。要执行全局迁移，只需指定源数据链路和目标数据链路。以下示例将 `ether0` 的所有 VNIC 迁移到 `net1`：

```
# dladm modify-vnic -l net1 -L ether0
```

其中

- `-l datalink` 指的是向其迁移 VNIC 的目标数据链路。
- `-L datalink` 指的是在其上配置 VNIC 的原始数据链路。

注 – 目标数据链路必须在源数据链路前面指定。

要执行选择性 VNIC 迁移，请指定要移动的 VNIC。以下示例将 `net0` 的选定 VNIC 迁移到 `net1`：

```
# dladm modify-vnic -l net1 vnic0,vnic1,vnic2
```

注 – `-L` 选项仅适用于全局修改。

在迁移一组 VNIC 时，还可以同时修改其 VLAN ID。但是，要指定新 VLAN ID，必须一次迁移一个 VNIC，如下例所示：

```
# dladm modify-vnic -l net1 -v 123 vnic0
# dladm modify-vnic -l net1 -v 456 vnic1
# dladm modify-vnic -l net1 -v 789 vnic2
```

迁移对 MAC 地址的影响取决于 VNIC 使用的是否为源数据链路的出厂 MAC 地址。

- 如果迁移过程中未指定 `-m` 选项，则在迁移后，出厂 MAC 地址将替换为目标数据链路中随机指定的地址。
- 如果迁移过程中使用了 `-m address` 选项，则迁移后会将该地址指定给 VNIC。

随机指定的 MAC 地址在迁移后不受影响并由其各自的 VNIC 保留。

以下示例介绍了如何迁移多个 VNIC。请注意，这些 VNIC 使用的是随机指定的 MAC 地址。因此，这些地址在迁移后不会发生更改。

```
# dladm show-vnic
LINK      OVER      SPEED  MACADDRESS      MACADDRTYPE  VID
vnic1     net0      1000   2:8:20:c2:39:38 random        0
vnic2     net0      1000   2:8:20:5f:84:ff random        0

# dladm modify-vnic -l net1 -L net0
# dladm show-vnic vnic0
LINK      OVER      SPEED  MACADDRESS      MACADDRTYPE  VID
vnic1     net1      1000   2:8:20:c2:39:38 random        0
vnic2     net1      1000   2:8:20:5f:84:ff random        0
```

显示 VNIC 信息

要获取有关您系统上 VNIC 的信息，请使用 `dladm show-vnic` 命令。

```
# dladm show-vnic
LINK      OVER      SPEED      MACADDRESS      MACADDRTYPE
vnic1     net0      1000 Mbps    2:8:20:c2:39:38 random
vnic2     net0      1000 Mbps    2:8:20:5f:84:ff random
```

VNIC 也是数据链路。因此，任何用于显示有关数据链路的信息的 `dladm` 命令也可用于显示有关系统上 VNIC（如果存在）的信息。例如，`dladm show-link` 在列表中包含 VNIC。或者，您可以使用 `dladm show-linkprop` 命令检查 VNIC 的属性。要获取有关某个 VNIC 的属性信息，请在显示链路属性时指定该 VNIC：

```
# dladm show-linkprop [-p property] vnic
```

▼ 如何删除 VNIC

此过程介绍了如何从系统中删除 VNIC 配置。以下步骤假设 VNIC 已连接到区域。您必须在全局区域中执行本过程。

- 1 因为 VNIC 已连接到区域，请停止该区域。

```
global# zoneadm -z zone halt
```

注 – 要确定区域所使用的链路，请使用 `dladm show-link` 命令。

- 2 从区域中删除或分离 VNIC。

```
global# zonecfg -z zone remove net physical=vnic
```

- 3 从系统中删除 VNIC。

```
global# dladm delete-vnic vnic
```

4 重新引导区域。

```
global# zonecfg -z zone boot
```

示例 2-5 从系统中删除 VNIC

在本示例中，将从 zoneB 和系统中删除 vnic1。

```
Global# dladm show-link
LINK          CLASS  MTU   STATE  OVER
net0          phys   1500  up     --
net2          phys   1500  up     --
net1          phys   1500  up     --
net3          phys   1500  up     --
zoneA/net0    vnic   1500  up     net0
zoneB/net0    vnic   1500  up     net0
vnic0         vnic   1500  up     net1
zoneA/vnic0   vnic   1500  up     net1
vnic1         vnic   1500  up     net1
zoneB/vnic1   vnic   1500  up     net1

Global# zoneadm -z zoneB halt
Global# zonecfg -z zoneB remove net physical=vnic1
Global# dladm delete-vnic vnic1
Global# zonecfg -z zoneB reboot
```


在 Oracle Solaris 中管理网络资源

本章介绍如何管理数据链路上的资源，包括虚拟链路，如 VNIC。网络资源管理可实现 IP 服务质量 (quality of service, QoS)，从而提高性能，尤其是虚拟网络的性能。

本章包含以下主题：

- 第 31 页中的“使用客户机、传送环和接收环”
- 第 39 页中的“使用池和 CPU”
- 第 43 页中的“管理流上的资源”

使用客户机、传送环和接收环

在 NIC 上，接收 (Rx) 环和传送 (Tx) 环是硬件资源，系统分别通过它们来接收和传送网络数据包。以下各节首先概述环，然后介绍用于为网络处理分配环的过程。还提供了示例，显示在发出命令分配环时环机制的工作方式。

MAC 客户机和环分配

MAC 客户机（如 VNIC 和其他数据链路）是在 NIC 上配置的，用于支持系统和其他网络节点之间的通信。配置客户机后，它同时使用 Rx 和 Tx 环来分别接收或传送网络数据包。MAC 客户机可以是基于硬件的或基于软件的。基于硬件的客户机需要满足以下任一条件：

- 有一个或多个专用 Rx 环。
- 有一个或多个专用 Tx 环。
- 有一个或多个专用 Rx 环和一个或多个专用 Tx 环。

不满足上述任一条件的客户机称为基于软件的 MAC 客户机。

根据具体的 NIC，可以为基于硬件的客户机指定专用环。一些 NIC（如 nxge）支持**动态环分配**。在此类 NIC 上，您不仅可以配置基于硬件的客户机，还可以灵活地确定要

分配给此类客户机的环的数量（假定环仍可供分配）。环的使用始终针对主接口进行优化，例如 `net0`。主接口也称为**主客户机**。任何尚未被其他客户机指定为可用的专用环会自动指定给主接口。

其他 NIC（如 `ixge`）仅支持**静态环分配**。对于这些 NIC，您只能创建基于硬件的客户机。这些客户机会自动配置为每个客户机有一组固定的环。NIC 驱动程序的初始配置过程中会确定固定的环组。有关静态环分配的驱动程序初始配置的更多信息，请参阅《[Oracle Solaris 11.1 可调参数参考手册](#)》。

基于软件的客户机没有专用环。而是与其他基于软件的现有客户机或主客户机共享环。基于软件的客户机所使用的环取决于在环分配中优先级高的基于硬件的客户机的数量。

了解**主客户机**和其他**辅助客户机**之间的差别很重要。主客户机是 NIC 的物理数据链路。基于安装过程中 Oracle Solaris 提供的通用名称，主客户机将被命名为 `netN`，其中，`N` 是实例号。有关数据链路的通用名称的说明，请参见《[Oracle Solaris 11 联网介绍](#)》中的“[网络设备和数据链路名称](#)”。VNIC 是在物理数据链路上创建的**辅助客户机**。如果这些客户机是基于硬件的客户机，它们可以有专用环。否则，这些客户机为基于软件的客户机。

VLAN 中的环分配

使用 VLAN 时，环分配根据创建 VLAN 的方式而有所不同。VLAN 通过以下两种方式之一创建：

- 使用 `dladm create-vlan` 子命令：

```
# dladm create-vlan -l link -v vid vlan
```
- 使用 `dladm create-vnic` 子命令：

```
# dladm create-vnic -l link -v vid vnic
```

使用 `dladm create-vlan` 子命令创建的 VLAN 与底层接口共享相同的 MAC 地址。因此，该 VLAN 还共享底层接口的 Rx 和 Tx 环。对于使用 `dladm create-vnic` 命令作为 VNIC 创建的 VLAN，其 MAC 地址与底层接口的不同。这种 VLAN 的环分配独立于底层链路的分配。因此，可以对该 VLAN 指定它自己的专用环（假定 NIC 支持基于硬件的客户机）。

环分配的数据链路属性

要管理环，可以通过使用 `dladm` 命令设置两个环属性：

- `rxrings` 指分配给指定链路的 Rx 环的数量。
- `txrings` 指分配给指定链路的 Tx 环的数量。

可以将每个属性设置为三个可能的值之一：

- `sw` 指示您正在配置基于软件的客户机。客户机没有专用环。相反，客户机与类似配置的任何其他现有客户机共享环。

- $n > 0$ （数大于零）仅适用于基于硬件的客户机的配置。数字指分配给客户机的专用环的数量。仅当底层 NIC 支持动态环分配时，才可以指定数量。
- `hw` 也适用于基于硬件的客户机的配置。然而，对于这种客户机，不能指定专用环的实际数量。相反，根据 NIC 驱动程序的初始配置，已经为每个客户机设置固定的环数。仅当底层 NIC 支持静态环分配时，才可以将 `*rings` 属性设置为 `hw`。

要提供有关当前环的指定和使用的信息，可以使用以下附加的只读环属性：

- `rxrings-available` 和 `txrings-available` 指可供分配的 Rx 和 Tx 环的数量。
- `rxhwcnt-available` 和 `txhwcnt-available` 指在 NIC 上可配置的 Rx 和 Tx 基于硬件的客户机的数量。

使用接收环和传送环的命令

要管理数据链路的接收环和传送环的使用，请使用以下基本 `dladm` 子命令：

- `dladm show-linkprop`—显示链路属性的当前值，包括 Rx 和 Tx 环。其输出提供有关数据链路的环支持功能的以下信息。您需要这些信息来确定可以配置哪种类型的客户机以使用 Rx 和 Tx 环。
 - 可以创建的可用客户机
 - 可以分配给可用客户机的可用环
 - 能否支持动态或静态环分配
 - 如果仅支持静态环分配，现有客户机的当前环分布

[第 32 页中的“环分配的数据链路属性”](#)进一步介绍了如何解释此命令的输出。

- `dladm show-phys -H datalink`—显示现有客户机当前正在如何使用物理数据链路的环。
- `dladm create-vnic -p ring-properties vnic`—创建具有特定 Tx 或 Rx 环数量的客户机来为通信服务。
- `dladm set-linkprop -p ring-properties datalink`—如果环可用且支持环分配，将环分配给特定客户机。

获取和解释环信息

本节介绍了显示数据链路与环相关的属性的 `dladm show-linkprop` 输出。

显示数据链路的环分配功能

本节提供了有关环相关属性的命令输出示例，并介绍了您可获取的信息类型。示例中使用了以下 NIC：

- `net0`（在 `nxge` 上）

- net1 (在 ixgbe 上)
- net2 (在 e1000g 上)

示例 3-1 nxge 环信息

以下示例显示了 nxge 的环信息：

```
# dladm show-linkprop net0
LINK      PROPERTY          PERM  VALUE  DEFAULT  POSSIBLE
...
net0      rxrings          rw    --    --        sw,<1-7>
...
net0      txrings          rw    --    --        sw,<1-7>
...
net0      rxrings-available r-    5     --        --
net0      txrings-available r-    5     --        --
net0      rxhwcCnt-available r-    2     --        --
net0      txhwcCnt-available r-    2     --        --
...
```

对于 net0，rxrings 和 txrings 的 POSSIBLE 字段的值为 sw 和 <1-7>。这些值指示 nxge 支持基于硬件的客户机以及基于软件的客户机。范围 <1-7> 指示您可以为客户机设置的 Rx 或 Tx 环的数量限制。该范围还指示 NIC 支持动态环分配同时用于接收端和传送端。

此外，*rings-available 属性指示有五个 Rx 环和五个 Tx 环可分配给基于硬件的客户机。

但是，*cCnt-available 属性显示您只能配置两个具有可用的专用 Rx 环的客户机。同样，您只能配置两个具有可用的专用 Tx 环的客户机。

示例 3-2 ixgbe 环信息

以下示例显示了 ixgbe 的环信息：

```
# dladm show-linkprop net1
LINK      PROPERTY          PERM  VALUE  DEFAULT  POSSIBLE
...
net1      rxrings          rw    --    --        sw,hw
...
net1      txrings          rw    --    --        sw,hw,<1-7>
...
net1      rxrings-available r-    0     --        --
net1      txrings-available r-    5     --        --
net1      rxhwcCnt-available r-    0     --        --
net1      txhwcCnt-available r-    7     --        --
...
```

对于 net1，rxrings 和 txrings 的 POSSIBLE 字段值 sw 和 hw 指示 ixgbe 同时支持基于硬件的客户机和基于软件的客户机。对于 Rx 环，仅支持静态环分配，即硬件将一组固定的 Rx 环指定给每个基于硬件的客户机。但是，对于 Tx 环，范围 <1-7> 指示支持动态分配。您可以确定将指定给基于硬件客户机的 Tx 环数量，在该示例中，最多七个环。

示例 3-2 ixgbe 环信息 (续)

此外，*rings-available 属性指示有五个 Tx 环可分配给基于硬件的客户机，但没有可指定的 Rx 环。

最后，根据 *hwcnt-available 属性，您可以配置七个基于硬件的 Tx 客户机专用 Tx 环。但是，您无法创建具有专用 Rx 环的基于硬件客户机，因为不支持动态 Rx 环分配。

任一 *rings-available 属性的 VALUE 字段下的零 (0) 表示以下各项之一：

- 再没有环可分配给客户机。
- 不支持动态环分配。

您可以通过将 rxrings 和 txrings 的 POSSIBLE 字段与 rxrings-available 和 txrings-available 的 VALUE 字段进行比较来验证零的含义。

例如，假定 txrings-available 为 0，如下所示：

```
# dladm show-linkprop net1
LINK  PROPERTY          PERM  VALUE  DEFAULT  POSSIBLE
...
net1  rxrings            rw    --    --        sw,hw
net1  txrings            rw    --    --        sw,hw,<1-7>
net1  rxrings-available r-    0      --    --
net1  txrings-available r-    0      --    --
...
```

在此输出中，rxrings-available 的 VALUE 字段为 0，而 rxrings 的 POSSIBLE 字段为 sw,hw。这些信息结合起来，意味着由于 NIC 不支持动态环分配而没有可用的 Rx 环。在传送端，txrings-available 的 VALUE 字段为 0，而 txrings 的 POSSIBLE 字段为 sw,hw,<1-7>。这些信息结合起来，表示由于所有 Tx 环均已分配而没有可用的 Tx 环。但是，txrings 的 POSSIBLE 字段表示支持动态环分配。因此，您可以在 Tx 环变得可用时分配这些环。

示例 3-3 e1000g 环信息

以下示例显示了 e1000g 的环信息：

```
# dladm show-linkprop net2
LINK  PROPERTY          PERM  VALUE  DEFAULT  POSSIBLE
...
net2  rxrings            rw    --    --        --
...
net2  txrings            rw    --    --        --
...
net2  rxrings-available r-    0      --    --
net2  txrings-available r-    0      --    --
net2  rxhwcnt-available r-    0      --    --
net2  txhwcnt-available r-    0      --    --
...
```

输出指示环和基于硬件的客户机均无法配置，因为 e1000g 中不支持环分配。

显示数据链路上的环使用和环分配

两个只读数据链路属性提供了有关数据链路上的现有客户机当前如何使用环的信息：

- rxrings-effective
- txrings-effective

要获取有关环使用以及将哪些环分布到客户机中的信息，请同时使用 `dladm show-linkprop` 和 `dladm show-phys -H` 子命令。

以下示例显示根据 Rx 和 Tx 环的使用以及这些环在客户机中的分布方式，这两个命令所生成的不同类型的输出。

示例 3-4 主客户机的环使用

主客户机是在 NIC 的物理数据链路上配置的接口。对于此示例，NIC 是 `ixgbe` 卡。缺省情况下，其数据链路是 `net0`。`net0` 上的 IP 接口是主客户机。

```
# dladm show-linkprop net0
LINK  PROPERTY          PERM  VALUE  DEFAULT  POSSIBLE
...
net0  rxrings           rw    --    --       sw,hw
net0  rxrings-effective r     2     --       --
net0  txrings           rw    --    --       sw,hw,<1-7>
net0  txrings-effective r     8     --       --
net0  txrings-available r-    7     --       --
net0  rxrings-available r-    0     --       --
net0  rxhwclnt-available r-    3     --       --
net0  txhwclnt-available r-    7     --       --
...
# dladm show-phys -H net0
LINK  RINGTYPE  RINGS  CLIENTS
net0  RX        0-1    <default,mcast>
net0  TX        0-7    <default>net0
net0  RX        2-3    net0
net0  RX        4-5    --
net0  RX        6-7    --
```

输出提供有关主客户机 `net0` 的环使用和环分布的下列信息：

- `rxrings-effective` 显示 `net0` 将自动接收两个 Rx 环。`txrings-effective` 显示 `net0` 使用八个 Tx 环。缺省情况下，所有未使用的环会自动指定给主客户机。
- 根据 `dladm show-phys -H` 命令的输出，分配给 `net0` 的两个 Rx 环为 2 和 3。对于 Tx 环，`net0` 使用环 0 到 7。

示例 3-5 辅助客户机的环使用

本示例假定在 `net0`（`ixgbe` 卡的物理数据链路）上配置了 VNIC 客户机 `vnic1`。

```
# dladm show-linkprop vnic1
LINK  PROPERTY          PERM  VALUE  DEFAULT  POSSIBLE
...
vnic1  rxrings           rw    hw     --       sw,hw
```

示例 3-5 辅助客户机的环使用 (续)

```

vnic1  rxrings-effective  r-   2   --   --
vnic1  txrings             rw   hw   --   sw, hw, <1-7>
vnic1  txrings-effective  r-   1   --   --
...
# dladm show-linkprop net0
LINK      PROPERTY              PERM  VALUE  DEFAULT  POSSIBLE
...
net0     rxrings                rw    --    --        sw, hw
net0     rxrings-effective     r-    2    --        --
net0     txrings               rw    --    --        sw, hw, <1-7>
net0     txrings-effective     r-    --    --        --
net0     txrings-available     r-    6    --        --
net0     rxrings-available     r-    0    --        --
net0     rxhwclnt-available   r-    3    --        --
net0     txhwclnt-available   r-    6    --        --
...
# dladm show-phys -H net0
LINK      RINGTYPE  RINGS      CLIENTS
net0     RX        0-1        <default,mcast>
net0     TX        0,2-7      <default>net0
net0     RX        2-3        net0
net0     RX        4-5        vnic1
net0     RX        6-7        --
net0     TX        1          vnic1

```

结合这三个命令的输出，可提供以下信息：

- vnic1 的 rxrings-effective 显示该 VNIC 将自动接收两个 Rx 环。txrings-effective 显示 vnic1 使用一个 Tx 环。按照为 *ring 属性设置的 hw 值所指示，这些环是静态分配的。
- 根据 dladm show-phys -H 命令的输出，分配给 net0 的两个 Rx 环为 2 和 3。对于 Tx 环，net0 使用环 0 和环 2 到 7。vnic1 为 Tx 环使用环 1，为 Rx 环使用环 4 和 5。

请注意，vnic1 配置为具有静态环分配的基于硬件的客户机。因此，net0 上可创建的可用 Tx 硬件客户机的数量 (txhwclnt-available) 减少为六个。

▼ 如何配置客户机并分配环

此过程介绍了如何基于环分配的支持类型在数据链路上配置客户机。请确保您可以解释显示数据链路环属性的 dladm 命令的输出，如第 33 页中的“显示数据链路的环分配功能”和第 36 页中的“显示数据链路上的环使用和环分配”中所述。这些信息可在您配置客户机的过程中给予指导。

1 显示数据链路的环属性。

```
# dladm show-linkprop datalink
```

从输出中确定以下信息：

- NIC 是否支持基于硬件的客户机

- NIC 支持的环分配的类型
- 要分配给基于硬件的客户机的环的可用性
- 您可以在链路上配置的基于硬件的客户机的可用性

2 根据前一个步骤所获得的信息，执行以下操作之一：

- 如果 NIC 支持动态环分配，请使用以下语法创建基于硬件的客户机：

```
# dladm create-vnic -p rxrings=number[,txrings=number] -l link vnic
```

如果之前创建了客户机，请使用以下语法：

```
# dladm set-linkprop -p rxrings=number[,txrings=number] vnic
```

注 - 一些 NIC 支持 Rx 环或 Tx 环上的动态环分配，但不同时支持这两种类型。您在支持动态环分配的环类型上指定 *number*。

- 如果 NIC 支持静态环分配，请使用以下语法创建基于硬件的客户机：

```
# dladm create-vnic -p rxrings=hw[,txrings=hw] -l link vnic
```

如果之前创建了客户机，请使用以下语法：

```
# dladm set-linkprop -p rxrings=hw[,txrings=hw] vnic
```

注 - 一些 NIC 支持 Rx 环或 Tx 环上的静态环分配，但不同时支持这两种类型。您在支持静态环分配的环类型上指定 *hw*。

- 如果 NIC 仅支持基于软件的客户机，请使用以下语法创建客户机：

```
# dladm create-vnic -p rxrings=sw[,txrings=sw] -l link vnic
```

如果之前创建了客户机，请使用以下语法：

```
# dladm set-linkprop -p rxrings=sw[,txrings=sw] vnic
```

3 (可选) 检查新创建的客户机的环信息。

```
# dladm show-linkprop vnic
```

4 (可选) 检查数据链路的环在不同客户机之间如何分布。

```
# dladm show-phys -H datalink
```

另请参见 有关显示如何使用流以及如何分配系统资源（包括 Rx 和 Tx 环）以在虚拟网络中处理网络通信的示例，请参见 [示例 3-8](#)。

使用池和 CPU

通过 `pool` 链路属性，您可将网络处理绑定到 CPU 池。使用此属性，您可以更好地整合区域中的网络资源管理与 CPU 绑定和管理。在 Oracle Solaris 中，区域管理包括使用 `zonecfg` 或 `poolcfg` 命令将非网络处理与 CPU 资源池绑定。要指定相同的资源池以便同时管理网络处理，请使用 `dladm set-linkprop` 命令来配置链路的 `pool` 属性。然后您将链路指定给该区域。

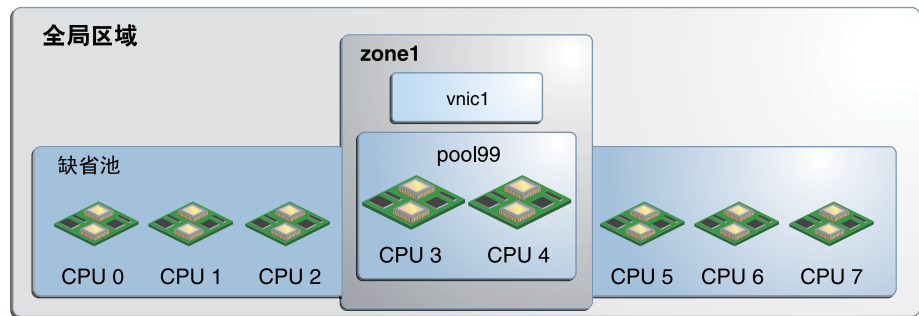
通过设置链路的 `pool` 属性并将该链路指定为一个区域的网络接口，该链路将同时绑定到该区域的池。如果该区域被设置为专用区域，则未指定给该区域的其他数据链路不再能够使用池中的 CPU 资源。

注 - 可以设置一个单独的属性 `cpu` 来将特定 CPU 指定给数据链路。`cpu` 和 `pool` 这两个属性是互斥的。您不能为给定数据链路同时设置这两个属性。要使用 `cpu` 属性将 CPU 资源指定给数据链路，请参见第 42 页中的“如何将 CPU 分配给链路”。

有关区域中池的更多信息，请参见《Oracle Solaris 11.1 管理：Oracle Solaris Zones、Oracle Solaris 10 Zones 和资源管理》中的第 13 章“创建和管理资源池（任务）”。有关创建池以及将 CPU 集指定给池的更多信息，请参阅 `poolcfg(1M)` 手册页。

下图显示将 `pool` 属性指定给数据链路时池的工作原理。

图 3-1 指定给一个区域的 VNIC 的 pool 属性



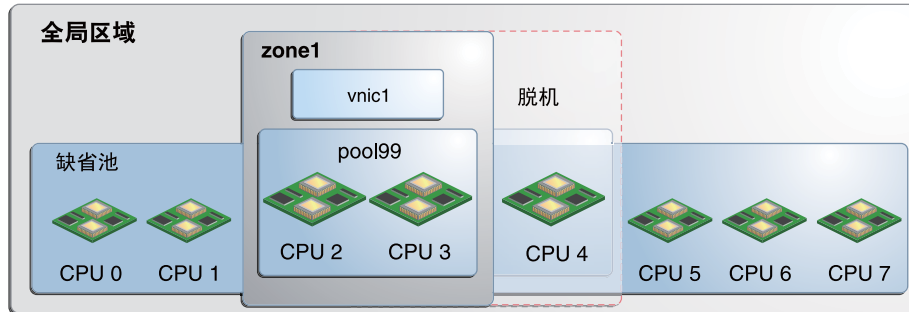
在图中，系统有八个 CPU。如果系统上没有配置池，则所有 CPU 都属于缺省池并由全局区域使用。不过，在本示例中，已创建 `pool99` 池，其中包含 CPU 3 和 CPU 4。该池与专用区域 `zone1` 相关联。如果 `pool99` 被设置为 `vnic1` 的一个属性，则 `pool99` 变为专用于管理 `vnic1` 的网络处理。在将 `vnic1` 指定为 `zone1` 的网络接口后，`pool99` 中的 CPU 将被保留，用于管理 `zone1` 的网络和非网络处理。

`pool` 属性在本质上是动态的。可以使用一个 CPU 范围配置区域池，内核确定将哪些 CPU 指定给该池的 CPU 集。对池的更改会自动针对数据链路实现，从而简化了该链路

的池管理。相反，通过使用 `cpu` 属性将特定 CPU 指定给链路时会要求您指定要分配的 CPU。每次要更改池的 CPU 组成时，您必须设置 `cpu` 属性。

例如，假定使图 3-1 中的系统 CPU 4 处于脱机状态。由于 `pool` 属性是动态的，软件会自动将另一个 CPU 与池相关联。因此，会保留池的两个 CPU 的初始配置。对于 `vnic1`，此更改是透明的。调整后的配置如下图所示。

图 3-2 pool 属性的自动重新配置



其他与池相关的属性显示有关数据链路的 CPU 使用或 CPU 池的信息。这些属性是只读属性，不能由管理员设置。

- `pool-effective` 显示正用于网络处理的池。
- `cpus-effective` 显示正用于网络处理的 CPU 的列表。

为了管理区域的 CPU 资源，设置数据链路的 `pool` 属性通常不作为初始步骤执行。更常见的是使用命令（如 `zonecfg` 和 `poolcfg`）将区域配置为使用资源池。`cpu` 和 `pool` 链路属性本身并未设置。在这种情况下，当引导区域时，会根据区域配置自动设置这些数据链路的 `pool-effective` 和 `cpus-effective` 属性。缺省池显示在 `pool-effective` 下，同时由系统选择 `cpus-effective` 的值。因此，如果使用 `dladm show-linkprop` 命令，则 `pool` 和 `cpu` 属性将为空，而 `pool-effective` 和 `cpus-effective` 属性将包含值。

要为网络处理绑定区域的 CPU 池，您还可以直接设置数据链路的 `pool` 和 `cpu` 属性。配置这些属性后，它们的值也会反映在 `pool-effective` 和 `cpus-effective` 属性中。但是请注意，通常不使用此替代步骤来管理区域的网络资源。

▼ 如何为数据链路配置 CPU 池

与其他链路属性一样，数据链路的 `pool` 属性可以在创建链路时设置，也可以稍后在链路需要进一步配置时设置。

要在创建 VNIC 时设置 `pool` 属性，请使用以下语法：

```
# dladm create-vnic -p pool=pool-name -l link vnic
```


要设置现有 VNIC 的 `pool` 属性，使用以下语法：

```
# dladm setlinkprop -p pool=pool-name vnic
```

以下过程介绍了如何为 VNIC 配置 CPU 池。

开始之前 您必须完成以下事项：

- 使用为其指定的 CPU 数量创建一个处理器集
- 创建一个将与处理器集相关联的池
- 将池与处理器集相关联

注 - 有关完成这些先决条件的说明，请参见《Oracle Solaris 11.1 管理：Oracle Solaris Zones、Oracle Solaris 10 Zones 和资源管理》中的“如何修改配置”。

- 1 将链路的 `pool` 属性设置为您为区域创建的 CPU 池。根据是否存在 VNIC，执行以下步骤之一：

- 如果尚未创建 VNIC，使用以下语法：

```
# dladm create-vnic -l link -p pool=pool vnic
```

其中 `pool` 指为区域创建的池的名称。

- 如果 VNIC 存在，请使用以下语法：

```
# dladm setlinkprop -p pool=pool vnic
```

- 2 将一个区域设置为使用 VNIC。

```
zonecfg>zoneid:net> set physical=vnic
```

注 - 有关解释如何将网络接口指定给区域的说明，请参阅《Oracle Solaris 11.1 管理：Oracle Solaris Zones、Oracle Solaris 10 Zones 和资源管理》中的“配置、检验并提交区域”

示例 3-6 将链路的 CPU 池指定给具有专用 IP 类型的区域

本示例说明如何将池指定给区域的数据链路。该方案基于图 3-1 中的配置。本示例假定已为区域配置一个名为 `pool99` 的 CPU 池。然后将该池指定给一个 VNIC。最后，将非全局区域 `zone1` 设置为使用该 VNIC 作为网络接口。

```
# dladm create-vnic -l net1 -p pool99 vnic1
```

```
# zonecfg -c zone1
zonecfg:zone1> set ip-type=exclusive
zonecfg:zone1> add net
```

```
zonecfg:zone1>net> set physical=vnic1
zonecfg:zone1>net> end
zonecfg:zone1> exit
```

▼ 如何将 CPU 分配给链路

以下过程说明如何通过配置 `cpu` 属性，指定特定 CPU 来处理数据链路上的通信流量。

1 检查接口的 CPU 指定。

```
# dladm show-linkprop -p cpus link
```

缺省情况下，没有将 CPU 指定给任何特定接口。因此，命令输出中的参数 `VALUE` 将不包含任何条目。

2 列出中断和与中断相关联的 CPU。

```
# echo ::interrupts | mdb -k
```

输出将列出系统上每个链路的参数，包括 CPU 数量。

3 将 CPU 指定给链路。

这些 CPU 可以包括与链路的中断相关联的 CPU。

```
# dladm set-linkprop -p cpus=cpu1,cpu2,... link
```

其中 `cpu1` 是要指定给该链路的 CPU 编号。您可以为链路指定多个专用 CPU。

4 检查链路中断来验证新的 CPU 指定。

```
# echo ::interrupts | mdb -k
```

5 (可选的) 显示与链路相关联的 CPU。

```
# dladm show-linkprop -p cpus link
```

示例 3-7 将 CPU 分配给链路

此示例说明如何将特定 CPU 指定给数据链路 `net0`。

请注意由不同的命令生成的输出中的以下信息。为清楚起见，重要信息在输出中强调显示。

- 缺省情况下，`net0` 没有专用 CPU。因此，`VALUE` 为 `--`。
- `net0` 的中断与 CPU 18 相关联。
- 分配 CPU 后，`net0` 在 `VALUE` 下显示新的 CPU 列表。

```
# dladm show-linkprop -p cpus net0
LINK          PROPERTY      PERM  VALUE      DEFAULT  POSSIBLE
net0          cpus          rw    --         --       --

# echo ::interrupts | mdb -k
```

```
Device Shared Type MSG # State INO Mondo Pil CPU
net#0 no MSI 2 enbl 0x1a 0x1a 6 18
```

```
# dladm set-linkprop -p cpus=14,18,19,20 net0
```

```
# dladm show-linkprop -p cpus net0
LINK PROPERTY PERM VALUE DEFAULT POSSIBLE
net0 cpus rw 14,18,19,20 -- --
```

所有支持线程（包括中断）现在只限于新指定的 CPU 集。

另请参见 有关显示如何使用流以及如何分配系统资源（包括 CPU 和 CPU 池）以在虚拟网络中处理网络通信的示例，请参见[示例 3-8](#)。

管理流上的资源

流包含根据一个属性组织的网络数据包。流使您能够进一步指定网络资源。有关流的概述，请参见第 12 页中的“通过使用流管理网络资源”。

使用流管理资源包含以下常规步骤：

1. 基于第 12 页中的“通过使用流管理网络资源”中列出的一个特定属性创建流。
2. 通过设置与网络资源相关的属性，定制流的资源使用。当前，仅可以设置用于处理数据包的带宽。

▼ 如何配置流

- 1 如果需要，列出可用链路以确定要在其上配置流的链路。

```
# dladm show-link
```

- 2 验证对所选链路上的 IP 接口正确配置了 IP 地址。

```
# ipadm show-addr
```

- 3 根据您为每个流确定的属性创建流。

```
# flowadm add-flow -l link -a attribute=value[,attribute=value] flow
```

link 指您在其上配置流的链路。

attribute 指以下分类之一（您可以根据这些分类将网络数据包组织到流中）：

- IP 地址
- 传输协议（UDP、TCP 或 SCTP）
- 应用程序的端口号（例如，端口 21 用于 FTP）

- DS 字段属性，仅用于 IPv6 数据包中的服务质量。有关 DS 字段的更多信息，请参阅《在 Oracle Solaris 11.1 中管理 IP 服务质量》中的“DS 代码点”。

flow 指为特定流指定的名称。

有关流和流属性的更多详细信息，请参见 `flowadm(1M)` 手册页。

4 (可选) 显示数据链路带宽值的可能范围。

```
# dladm show-linkprop -p maxbw link
```

其中，*link* 是在其上配置流的数据链路。

POSSIBLE 字段中列出了值的范围。

5 向流分配带宽份额。

```
# flowadm set-flowprop -p maxbw=value flow
```

您设置的值必须在链路带宽值的允许范围内。

注 - 当前，仅可以定制一个流的带宽。

6 (可选) 显示链路上已创建的流。

```
# flowadm
```

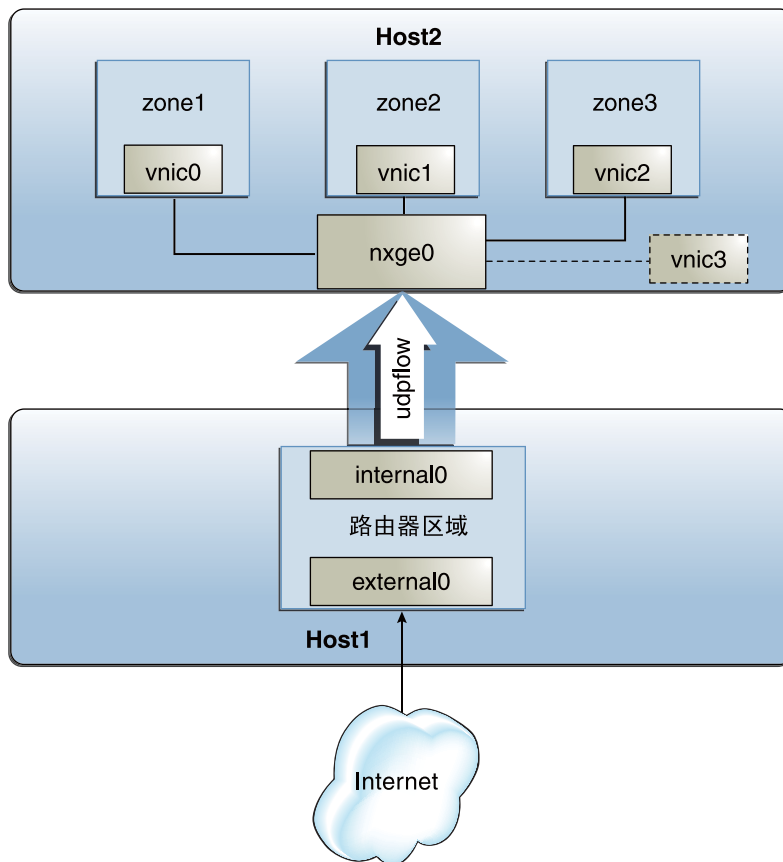
注 - 如果使用 `flowadm` 命令时不带任何子命令，该命令将提供与 `flowadm show-flow` 命令相同的信息。

7 (可选) 显示指定流的属性值。

```
# flowadm show-flowprop flow
```

示例 3-8 通过设置链路和流属性管理资源

本示例将为数据链路和流指定网络资源的步骤合并在一起。本示例基于下图中所示的配置。



图中显示了两个相互连接的物理主机。

- Host1 具有以下配置：
 - 它有一个非全局区域，充当路由器区域。为该区域指定了两个接口：`net0` 连接到 Internet，`net1` 连接到包含第二个主机的内部网络。
 - `udpflow` 是在 `net0` 上配置的流，用于隔离 UDP 通信并对 UDP 包使用资源的方式实施控制。有关配置流的信息，请参见第 43 页中的“管理流上的资源”。
- Host2 具有以下配置：
 - 它有三个非全局区域及其各自的 VNIC。这些 VNIC 是在 `net0` 上配置的，该接口的卡支持动态环分配。有关环分配的更多信息，请参见第 31 页中的“使用客户机、传送环和接收环”。
 - 每个区域的网络处理负载是不同的。在本示例中，`zone1` 的负载重，`zone2` 的负载为中等，`zone3` 的负载轻。根据这些区域的负载情况对其指定资源。

- 将一个单独的 VNIC 配置为基于软件的客户机。有关 MAC 客户机的概述，请参见第 31 页中的“MAC 客户机和环分配”。

本示例中的任务涉及以下方面：

- 创建流和配置流控制—在 net1 上创建了一个流，以便对 Host2 接收的 UDP 包进行单独的资源控制。
- 为 Host2 上的 VNIC 配置网络资源属性—根据每个区域上的处理负载，为每个区域的 VNIC 配置一组专用环。还要配置一个单独的没有专用环的 VNIC 作为基于软件的客户机的示例。

请注意，该示例不包括任何区域配置过程。要配置区域，请参阅《Oracle Solaris 11.1 管理：Oracle Solaris Zones、Oracle Solaris 10 Zones 和资源管理》中的第 17 章“规划和配置非全局区域（任务）”。

首先，查看有关 Host1 上的链路和 IP 接口的信息。

```
# ipadm
NAME                CLASS/TYPE STATE   UNDER   ADDR
lo0                  loopback  ok      --      --
  lo0/v4              static    ok      --      127.0.0.1/8
net0                  ip        ok      --      --
  net0/v4              static    ok      --      10.10.6.5/24
net1                  ip        failed  ipmp0   --
  net1/v4              static    ok      --      10.10.12.42/24
```

接下来，在 net1 上创建一个流以将 UDP 通信隔离到 Host2。然后，实现该流上的资源控制。

```
# flowadm add-flow -l net1 -a transport=udp udpflow
# flowadm set-flowprop -p maxbw=80 udpflow
```

然后，检查有关创建的流的信息。

```
flowadm
FLOW      LINK  IPADDR  PROTO  LPORT  RPORT  DFSLD
udpflow   net1  --      udp    --     --     --

# flowadm show-flowprop
FLOW      PROPERTY  VALUE  DEFAULT  POSSIBLE
udpflow   maxbw     80     --       --
```

在 Host2 上，为每个区域在 net0 上配置 VNIC。在每个 VNIC 上实现资源控制。然后，将 VNIC 指定给其各自的区域。

```
# dladm create-vnic -l net0 vnic0
# dladm create-vnic -l net0 vnic1
# dladm create-vnic -l net0 vnic2

# dladm set-prop -p rxrings=4,txrings=4 vnic0
# dladm set-prop -p rxrings=2,txrings=2 vnic1
```

```
# dladm set-prop -p rxrings=1,txrings=1 vnic2

# zonecfg -z zone1
# zonecfg:zone1> add net
# zonecfg:zone1:net> set physical=vnic0
# zonecfg:zone1:net> end
# zonecfg:zone1> commit
# zonecfg:zone1> exit
# zoneadm -z zone1 reboot

# zonecfg -z zone2
# zonecfg:zone2> add net
# zonecfg:zone2:net> set physical=vnic1
# zonecfg:zone2:net> end
# zonecfg:zone2> commit
# zonecfg:zone2> exit
# zoneadm -z zone2 reboot
#

# zonecfg -z zone3
# zonecfg:zone3> add net
# zonecfg:zone3:net> set physical=vnic2
# zonecfg:zone3:net> end
# zonecfg:zone3> commit
# zonecfg:zone3> exit
# zoneadm -z zone3 reboot
#
```

假定 pool1 (Host2 中的一个 CPU 集) 以前配置为由 zone1 使用。绑定该 CPU 池以便同时管理 zone1 的网络处理, 如下所示:

```
# dladm set-prop -p pool=pool1 vnic0
```

最后, 创建一个基于软件的客户机, 它与主接口 net0 共享环。

```
# dladm create-vnic -p rxrings=sw,txrings=sw -l net0 vnic3
```


监视 Oracle Solaris 中的网络通信和资源使用情况

本章介绍用于监视和收集有关物理和虚拟网络环境 Oracle Solaris 11 中网络资源使用统计数据的任务。这些信息可以帮助您分析资源分配以便置备、整合和记帐。本章介绍了用于显示统计数据两个命令：`d1stat` 和 `flowstat`。

讨论以下主题：

- 第 49 页中的“网络通信流概述”
- 第 52 页中的“用于监视通信统计数据的命令”
- 第 52 页中的“收集有关链路的网络通信统计数据”
- 第 56 页中的“收集有关流的网络通信统计数据”
- 第 57 页中的“为网络通信配置网络记帐”

网络通信流概述

数据包在流入或流出系统时会遍历一个路径。准确地说，数据包分别通过 NIC 的接收 (Rx) 环和传送 (Tx) 环接收和传送。通过这些环，接收到的包向上传递到网络栈以供进一步处理，而传出包被发送到网络。

本节介绍了网络通道的概念。**网络通道**是为管理网络通信而分配的系统资源组合。因此，网络通道是针对特定类型的网络通信的定制路径。每个通道要么是**硬件通道**，要么是**软件通道**。此外，每个通道类型要么是**接收通道**，要么是**传送通道**。

硬件和软件通道的区别在于 NIC 支持环和环分配的能力，如第 31 页中的“使用客户机、传送环和接收环”中所述。本章重点讲述通过接收通道接收的传入通信。

注 - Rx 和 Tx 环以及其他网络资源通过设置数据链路的属性来指定。因此，数据链路是系统上的网络通道。

在硬件通道中，指定给这些通道的环为包所专用。而在软件通道中，所有网络包都共享这些通道中的环。将数据链路配置为共享环的原因如下：

- 便于管理。数据链路可能没有执行高负载处理，因此不需要专用环。
- NIC 不支持环分配。
- 尽管支持环分配，但环不再可用于指定为专用环。

下图显示了不同的硬件通道。

图 4-1 硬件通道

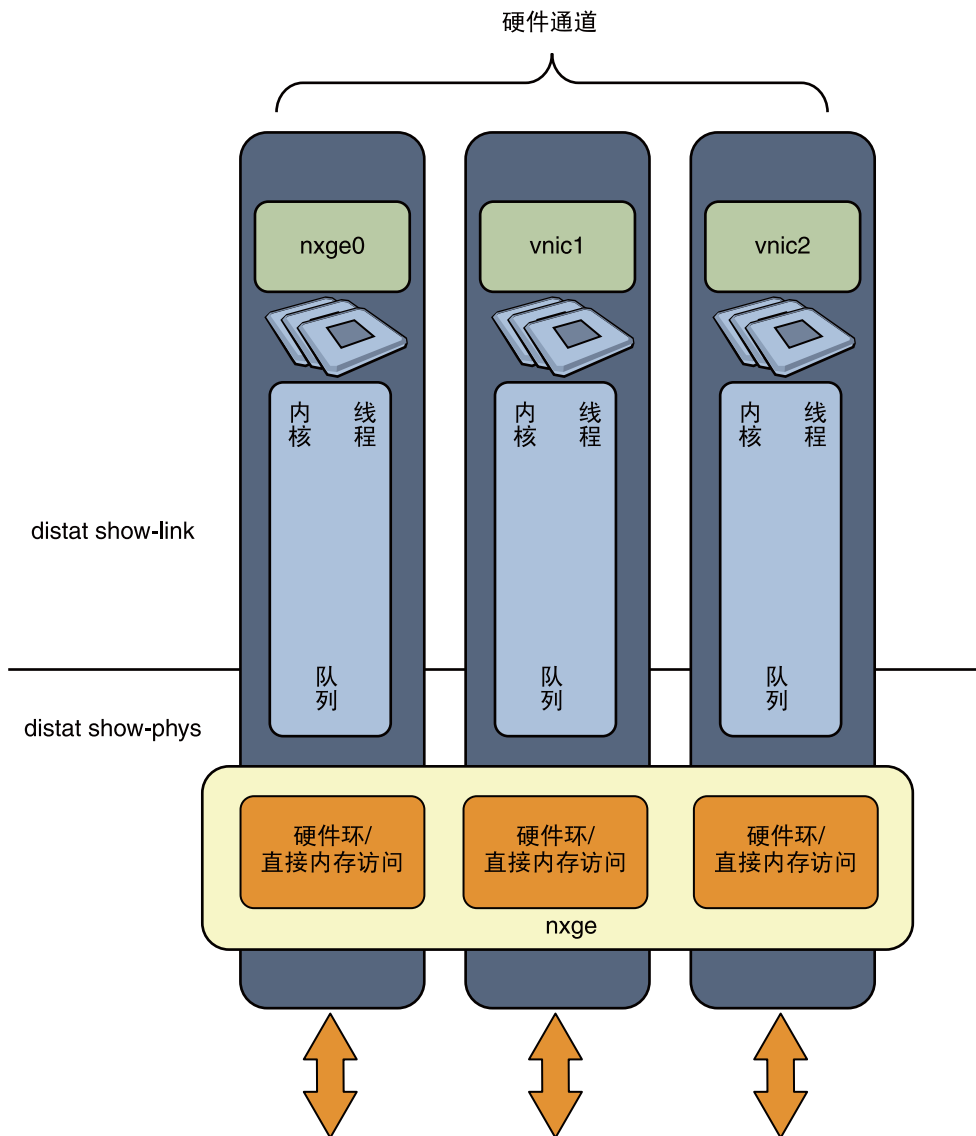


图 4-1 显示了以下配置：

- 在 `net0` 上配置了两个 VNIC： `vnic1` 和 `vnic2`。 `net0` 是主客户机，也是主通道。 VNIC 是辅助客户机，也是网络通道。
- 将多组硬件环指定给辅助客户机。每台客户机（包括主客户机）都可充当硬件通道。

- 为每个通道分配了一组 CPU。

后续各节将介绍如何监视流经这些通道的通信。

用于监视通信统计数据的命令

通过 `dlstat` 和 `flowstat` 命令，您可以分别监视和获取有关数据链路和流的网络通信统计数据。以上命令等效于 `dladm` 和 `flowadm` 命令。下表将一对 `*adm` 命令与一对 `*stat` 命令的功能进行了比较。

管理命令		监视命令	
命令	功能	命令	功能
<code>dladm</code> 命令选项	用于配置和管理数据链路的用户界面和工具	<code>dlstat</code> 命令选项	用于获取有关数据链路的通信统计数据的用户界面和工具
<code>flowadm</code> 命令选项	用于配置和管理流的用户界面和工具	<code>flowstat</code> 命令选项	用于获取有关流的通信统计数据的用户界面和工具

后续几节将对各个监视命令进行更详细的介绍。

收集有关链路的网络通信统计数据

您可以使用以下 `dlstat` 命令的变体收集网络通信信息。

命令	提供的信息
<code>dlstat [link]</code> <code>dlstat -rt [link]</code> <code>dlstat show-link [link]</code>	每个通道的传入和传出通信统计数据
<code>dlstat show-link -rt [link]</code>	每个通道每个环的传入和传出通信统计数据
<code>dlstat show-phys [link]</code>	每个网络物理设备的传入和传出通信统计数据
<code>dlstat show-phys -rt [link]</code>	每个网络物理设备每个环的传入和传出通信统计数据
<code>dlstat show-aggr [link]</code> <code>dlstat show-aggr -rt [link]</code>	每个聚合每个端口的传入和传出通信统计数据

您可以分别在 `dlstat` 命令中使用 `-r` 选项或 `-t` 选项来只显示接收端或传送端的统计信息。此外，您还可以将其他选项与 `dlstat` 命令一起使用。有关更多信息，请参阅 `dlstat(1M)` 手册页。

获取网络设备的网络通信统计数据

`dlstat show-phys` 子命令可提供有关物理网络设备的统计数据。如图 4-1 所示，该子命令作用于硬件环（位于网络栈的设备层）。等效子命令 `dladm show-phys` 也作用于栈的同一层。请将图 4-1 与《Oracle Solaris 11 联网介绍》中的“Oracle Solaris 11 的实现”中说明的网络栈相比较。

以下示例显示有关系统上所有物理链路的统计数据：

```
# dlstat show-phys
LINK  IPKTS  RBYTES  OPKTS  OBYTES
net0  2.14M  257.48M  3.19M  210.88M
net1  1.15M  120.32M  1.00M  98.70M
net2  1.10M  110.10M  1.28   183.00M
...
```

该输出显示了系统每个链路的传入和传出通信的统计数据。包的数量和每个包的字节大小已显示。

以下示例显示了 `net0` 的每个硬件环的接收端统计数据：

```
# dlstat show-phys -r net0
LINK  TYPE  ID  INDEX  IPKTS  RBYTES
net0  rx    local  --    0      0
net0  rx    hw    1      0      0
net0  rx    hw    2    1.73M  2.61G
net0  rx    hw    3      0      0
net0  rx    hw    4    8.44M  12.71G
net0  rx    hw    5    5.68M  8.56G
net0  rx    hw    6    4.99M  7.38G
net0  rx    hw    7      0      0
```

第二个输出显示，`net0` 设备有八个接收环，这些环在 `INDEX` 字段下标识。在每个环上平均分布数据包是一种理想配置，表示根据链路的负载将环适当地分配到了链路。不均匀的分布表示每个链路的环分布不成比例。不均匀分布的解决方法取决于 NIC 是否支持动态环分配。如果 NIC 支持动态环分配，您可以重新分布每个链路上的环以均衡包处理负载。有关动态环分配的更多信息，请参见第 31 页中的“使用客户机、传送环和接收环”。

以下示例显示了有关设备每秒接收的通信的信息。使用 `-i` 选项可指定时间间隔。要停止刷新显示，按 `Ctrl-C`。

```
# dlstat show-phys -r -i 1
LINK  TYPE  INDEX  IPKTS  RBYTES
net0  rx    0    101.91K  32.86M
```

```

net1  rx      0      9.61M   14.47G
net2  rx      8      336K    0
net0  rx      0        0      0
net1  rx      0     82.13K  123.69M
net2  rx      0        0      0
...
^C

```

本示例显示了 net1 作为网络设备时其传送环的使用情况。

```

# dlstat show-phys -t net1
LINK TYPE INDEX  OPKTS  OBYTES
net1  tx      0       44    3.96K
net1  tx      1        0      0
net1  tx      2    1.48M  121.68M
net1  tx      3    2.45M  201.11M
net1  tx      4    1.47M  120.82M
net1  tx      5        0      0
net1  tx      6    1.97M  161.57M
net1  tx      7    4.59M  376.21M
net1  tx      8    2.43M  199.24M
net1  tx      9        0      0
net1  tx     10    3.23M  264.69M
net1  tx     11    1.88M  153.96M

```

获取通道的网络通信统计数据

dlstat show-link 子命令提供了有关在物理链路上配置的通道的统计数据。这些通道由数据链路层构成。如图 4-1 中所示，该子命令作用于网络栈的数据链路层。等效子命令 dladm show-link 也作用于栈的同一层。请将图 4-1 与《Oracle Solaris 11 联网介绍》中的“Oracle Solaris 11 的实现”中说明的网络栈相比较。

以下示例显示了 vnic0 的接收端通信统计数据。

```

# dlstat show-link -r vnic0
LINK TYPE  ID  INDEX  IPKTS  RBYTES  INTRS  POLLS  IDROPS
vnic0  rx  hw     2    1.73M   2.61G   1.33M  400.22K  0
vnic0  rx  hw     4    8.44M  12.71G   4.35M  4.09M   0

```

先前的输出显示了通道 vnic0 的通信统计数据。为该通道分配了两个接收环（环 2 和环 4）供通道专用。该输出显示了这两个环用于传入网络通信的情况。但是，数据还可能反映其他资源分配的实现方式，如带宽限制和设置优先级。

假设为主通道 net0 显示了以下信息：

```

# dlstat show-link -r net0
LINK TYPE  ID  INDEX  IPKTS  RBYTES  INTRS  POLLS  IDROPS
net0  rx  local  --      0      0      0      0      0
net0  rx  sw    --  794.28K  1.19G  794.28K  0      0
...

```

根据该输出可以看出，当前在与其他客户机共享其中一个 Rx 环（环 0）。如果没有为辅助客户机配置分配的环，则将共享环。不分配环的可能原因如下：

- 无法再在链路上创建硬件客户机。
- 无法再分配硬件环。
- 管理员有意配置软件客户机。

有关中断 (INTRS) 和丢包 (*DROPS) 的统计数据同样十分重要。中断次数少和零丢包表示性能效率高。如果中断次数或者丢包次数很高，您可能需要向通道添加更多的资源。

以下示例显示了有关主通道 net1 所使用的环上的传出包的统计数据。该输出显示 net1 使用了所有的 Tx 环。

```
# dlstat show-link -t net1
LINK  TYPE  ID  INDEX  OPKTS  OBYTES  ODROPS
net1  tx    hw   0      32     1.44K   0
net1  tx    hw   1       0       0       0
net1  tx    hw   2    1.48M   97.95M  0
net1  tx    hw   3    2.45M  161.87M 0
net1  tx    hw   4    1.47M   97.25M  0
net1  tx    hw   5       0      276     0
net1  tx    hw   6    1.97M  130.25M  0
net1  tx    hw   7    4.59M  302.80M  0
net1  tx    hw   8    2.43M  302.80M  0
net1  tx    hw   9       0       0       0
net1  tx    hw  10    3.23M  213.05M  0
net1  tx    hw  11    1.88M  123.93M  0
```

以下命令显示了有关链路 net1 的接收端使用情况的统计数据。此外，在命令中使用 -F 选项时，输出还提供扇出信息。具体来说，扇出计数有两个 (0 和 1)。使用环 0 的硬件通道上接收的网络通信被分流并传递到两个扇出端。同样，使用环 1 的硬件通道上接收的网络通信也被分流到两个扇出端。

```
# dlstat show-link -r -F net1
LINK  ID  INDEX  FOUT  IPKTS
net1  local --      0      0
net1  hw   0      0  382.47K
net1  hw   0      1      0
net1  hw   1      0  367.50K
net1  hw   1      1  433.24K
```

获取链路聚合的网络通信统计数据

dlstat show-aggr 命令显示了当通信遍历系统上的聚合时，每个聚合端口的网络包统计数据。

```
# dlstat show-aggr
LINK  PORT  IPKTS  RBYTES  OPKTS  OBYTES
aggr1  --      0      0      0      0
aggr1  net0   0      0      0      0
aggr1  net1   0      0      0      0
```

该输出指示了带有两个底层链路 (即 net0 和 net1) 的链路聚合 aggr1 的配置。由于网络通信是系统通过聚合接收或发送的，因此将报告每个端口的有关传入和传出包以及它们各自大小的信息。端口通过聚合的底层链路标识。

收集有关流的网络通信统计数据

流统计数据有助于评估系统上定义的任何流的数据包通信。要获取流信息，请使用 `flowstat` 命令。有关此命令的更多信息，请参阅 [flowstat\(1M\)](#) 手册页。

`flowstat` 命令最常用的语法如下：

```
# flowstat [-r|-t] [-i interval] [-l link] [flow]
```

`[-r|-t]` 只显示接收端统计数据（`-r` 选项）或只显示传送端统计数据（`-t` 选项）。要同时显示有关接收端和传送端的统计数据，请省略该选项。

`-i interval` 指定刷新显示统计数据的时间间隔（以秒为单位）。如果不使用此选项，则显示静态输出。

`-l link` 指示要监视指定数据链路上所有流的统计数据。如果不使用此选项，则显示有关所有数据链路上所有流的信息。

`flow` 指示要仅监视指定流的统计数据。如果不使用此选项，则根据您是否指定了一个链路，显示所有流统计数据。

以下示例说明了显示系统中已配置流的信息的不同方法。

示例 4-1 显示所有流的通信统计数据（时间间隔为一秒）

本示例显示有关系统上配置的所有流的传入和传出通信的信息（时间间隔为每秒）。

```
# flowstat -i 1
FLOW      IPKTS      RBYTES      IERRS      OPKTS      OBYTES      OERRS
flow1     528.45K    787.39M      0           179.39K    11.85M      0
flow2     742.81K    1.10G        0           0           0           0
flow3      0           0            0           0           0           0
flow1     67.73K    101.02M      0           21.04K     1.39M      0
flow2      0           0            0           0           0           0
flow3      0           0            0           0           0           0
...
^C
```

本示例显示有关配置的所有流的传出通信的统计数据。

示例 4-2 显示所有流的传送端统计数据

```
# flowstat -t
FLOW      OPKTS      OBYTES      OERRS
flow1     24.37M     1.61G      0
flow2      0           0           0
flow1      4           216        0
```

示例 4-3 显示指定链路上所有流的接收端统计数据

本示例显示在数据链路 `net0` 上创建的所有流的硬件通道中的传入通信。

示例 4-3 显示指定链路上所有流的接收端统计数据 (续)

```
# flowstat -r -i 2 -l net0
FLOW      IPKTS    RBYTES    IERRS
tcp-flow  183.11K  270.24M    0
udp-flow   0        0          0
tcp-flow  373.83K  551.52M    0
udp-flow   0        0          0
tcp-flow  372.35K  549.04M    0
udp-flow   0        0          0
tcp-flow  372.87K  549.61M    0
udp-flow   0        0          0
tcp-flow  371.57K  547.89M    0
udp-flow   0        0          0
tcp-flow  191.92K  282.95M    0
udp-flow  206.51K  310.70M    0
tcp-flow   0        0          0
udp-flow  222.75K  335.15M    0
tcp-flow   0        0          0
udp-flow  223.00K  335.52M    0
tcp-flow   0        0          0
udp-flow  160.22K  241.07M    0
tcp-flow   0        0          0
udp-flow  167.89K  252.61M    0
tcp-flow   0        0          0
udp-flow   9.52K   14.32M    0
^C
```

为网络通信配置网络记帐

您可以使用扩展记帐功能在系统上设置网络记帐。网络记帐涉及在日志文件中捕获有关网络通信的统计数据。通过这种方式，您可以维护通信的记录以便跟踪、置备、整合和记帐。稍后，您可以查看日志文件获取有关一段时间内网络使用情况的历史信息。

要设置网络记帐，请使用扩展记帐功能的 `acctadm` 命令。设置完网络记帐之后，请使用 `flowstat` 命令记录通信统计数据。

本节介绍了以下过程：

- 第 58 页中的“如何设置网络记账”
- 第 59 页中的“如何获取有关网络通信的历史统计数据”

▼ 如何设置网络记帐

- 1 在具有您要跟踪其网络使用情况的接口的系统上，成为管理员。

有关更多信息，请参见《Oracle Solaris 11.1 管理：安全服务》中的“如何使用指定给您的管理权限”。

- 2 查看扩展记帐功能可以支持的记帐类型的状态。

```
# acctadm [process | task | flow | net]
```

扩展记帐功能可以支持四种记帐类型。acctadm 命令的可选操作数对应于这些记帐类型。您可以使用带操作数的该命令配置特定类型的记帐。

- 进程记帐
- 任务记帐
- IPQoS 的流记帐
- 网络记帐链路和流

注 - 网络记帐也适用于由第 43 页中的“管理流上的资源”中所述的 flowadm 和 flowstat 命令管理的流。因此，要设置这些流的记帐，请结合使用 net 选项和 acctadm 命令。请勿使用 flow 选项，该选项启用 IPQoS 配置的流记帐。

指定 net 会显示网络记帐的状态。如果不使用 net，则显示所有四种记帐类型的状态。

- 3 为网络通信启用扩展记帐。

```
# acctadm -e extended -f filename net
```

其中 *filename* 包括用于捕获网络通信统计数据的日志文件的完整路径。可以在您指定的任何目录中创建日志文件。

- 4 验证已激活的扩展网络记帐。

```
# acctadm net
```

示例 4-4 在系统中设置网络记帐

本示例说明如何在系统中配置网络记帐以捕获和显示通信历史信息。

首先，查看所有记帐类型的状态，如下所示：

```
# acctadm
      Task accounting: inactive
      Task accounting file: none
      Tracked task resources: none
      Untracked task resources: extended
      Process accounting: inactive
      Process accounting file: none
```

```

Tracked process resources: none
Untracked process resources: extended,host
    Flow accounting: inactive
    Flow accounting file: none
Tracked flow resources: none
Untracked flow resources: extended
    Network accounting: inactive
    Network accounting file: none
Tracked Network resources: none
Untracked Network resources: extended

```

输出显示网络记帐未处于活动状态。

接下来，启用扩展网络记帐。

```

# acctadm -e extended -f /var/log/net.log net
# acctadm net
    Net accounting: active
    Net accounting file: /var/log/net.log
Tracked net resources: extended
Untracked net resources: none

```

▼ 如何获取有关网络通信的历史统计数据

启用网络记帐之后，您可以使用 `dlstat` 和 `flowstat` 命令从日志文件中提取信息。此过程介绍了这些步骤。

开始之前 您必须为网络启用扩展记帐，才能显示有关网络的历史数据。此外，要显示有关流的通信历史数据，必须先按第 43 页中的“[管理流上的资源](#)”中所述配置系统中的流。

1 在具有您要跟踪其网络使用情况的接口的系统上，成为管理员。

有关更多信息，请参见《[Oracle Solaris 11.1 管理：安全服务](#)》中的“[如何使用指定给您的管理权限](#)”。

2 要提取和显示有关数据链路的资源使用情况历史信息，请使用以下命令：

```
# dlstat show-link -h [-a] -f filename [-d date] [-F format] [-s start-time] [-e end-time] [link]
```

-h	按照数据链路的传入和传出包，显示有关资源使用情况的历史信息摘要。
-a	显示所有数据链路的资源使用情况，包括那些在数据捕获后已被删除的数据链路。
-f filename	指定在使用 <code>acctadm</code> 命令启用网络记帐时定义的日志文件。
-d date	显示指定日期的记录信息。
-F format	以特定格式显示数据，以便随后可以对其进行绘图分析。当前， <code>gnuplot</code> 是唯一受支持的格式。

- s *start-time* \
 - e *end-time* 显示指定的日期和时间范围内可用的记录信息。使用 MM/DD/YYYY, hh:mm:ss 格式。hour (hh) 必须使用 24 小时制表示法。如果未包含日期，则将显示对应于当前日期在指定时间范围内的数据。
 - link* 显示指定数据链路的历史数据。如果不使用此选项，则显示所有配置的数据链路的历史网络数据。
- 3 要提取并显示有关配置的流的网络通信历史信息，请使用以下命令：**
- ```
flowstat -h [-a] -f filename [-d date] [-F format] [-s start-time] [-e end-time] [flow]
```
- h                    按照配置的流的传入和传出包，显示有关资源使用情况的历史信息摘要。
  - a                    显示所有配置的流的资源使用情况，包括那些在数据捕获后已被删除的配置流。
  - f *filename*        指定在使用 `acctadm` 命令启用网络记帐时定义的日志文件。
  - d                    显示指定日期的记录信息。
  - F *format*           以特定格式显示数据。当前，`gnuplot` 是唯一支持的格式。
  - s *start-time* \
  - e *end-time*        显示指定的日期和时间范围内可用的记录信息。使用 MM/DD/YYYY, hh:mm:ss 格式。小时 (hh) 必须使用 24 小时制表示法。如果未包含日期，则将显示对应于当前日期在指定时间范围内的数据。
  - flow*                显示指定流的历史数据。如果不使用此选项，则显示所有配置的流的历史网络数据。

#### 示例 4-5 显示有关数据链路的资源使用情况历史信息

以下示例显示了有关指定数据链路的网络通信及其资源使用的历史统计数据：

```
dlstat show-link -h -f /var/log/net.log net0
LINK DURATION IPACKETS RBYTES OPACKETS OBYTES BANDWIDTH
net0 80 1031 546908 0 0 2.44 Kbps
```

#### 示例 4-6 显示有关流的资源使用情况历史信息

以下示例说明了显示有关流的网络通信及其资源使用的历史统计数据的不同方法。

以下示例显示了有关流的通信资源使用情况的历史统计数据：

```
flowstat -h -f /var/log/net.log
FLOW DURATION IPACKETS RBYTES OPACKETS OBYTES BANDWIDTH
flowtcp 100 1031 546908 0 0 43.76Kbps
flowudp 0 0 0 0 0 0.00Mbps
```

以下示例显示了在给定的日期和时间范围内有关流的通信资源使用情况的历史统计数据：

```
flowstat -h -s 02/19/2008,10:39:06 -e 02/19/2008,10:40:06 \
-f /var/log/net.log flowtcp

FLOW START END RBYTES OBYTES BANDWIDTH
flowtcp 10:39:06 10:39:26 1546 6539 3.23 Kbps
flowtcp 10:39:26 10:39:46 3586 9922 5.40 Kbps
flowtcp 10:39:46 10:40:06 240 216 182.40 bps
flowtcp 10:40:06 10:40:26 0 0 0.00 bps
```

以下示例显示了在给定的日期和时间范围内有关流的通信资源使用情况的历史统计数据。此信息是使用 gnuplot 格式显示的。

```
flowstat -h -s 02/19/2008,10:39:06 -e 02/19/2008,10:40:06 \
-F gnuplot -f /var/log/net.log flowtcp
Time tcp-flow
10:39:06 3.23
10:39:26 5.40
10:39:46 0.18
10:40:06 0.00
```



# 索引

---

## A

acctadm 命令, 57-61

## C

传输协议, 12

CPU, 13, 39

    CPU 分配, 42-43

CPU 池, 12, 39

    缺省池, 39

    指定给链路, 40

## D

dladm 命令

    create-etherstub, 10, 16

    create-vlan, 32

    create-vnic, 10-11, 12, 16

    delete-vnic, 28-29

    modify-vnic, 25

    set-linkprop, 11, 12

    show-linkprop, 33

    show-phys, 33

    show-vnic, 28

dlstat 命令, 49, 52

    show-aggr, 55

    show-link, 54-55

    show-phys, 53-54

DS 字段, 12

## E

etherstub, 8-9, 15-16

    创建, 10, 16

    专用虚拟网络, 22-24

## F

flowadm 命令, 43-47

    add-flow, 13, 43

    help, 13

    set-flowprop, 13, 44

    show-flowprop, 44

flowstat 命令, 49, 52, 56-57

## G

GARP VLAN 注册协议 (GARP VLAN Registration Protocol, GVRP), 17

## I

ipadm 命令

    create-addr, 16

    create-ip, 16

    显示 IP 信息, 46

## M

MAC 地址, 在 VNIC 上, 8

MAC 客户机, 31  
  分配环, 37-38  
  辅助, 32, 36  
  基于软件, 31  
  基于硬件, 31  
  配置, 37-38  
  主, 31, 36

## N

NIC 环, 12  
  另请参见环分组

## S

SCTP, 12

## T

TCP, 12

## U

UDP, 12

## V

VLAN  
  修改 VNIC 的 VLAN ID, 25-26  
  作为 VNIC, 16-17  
VNIC, 7-11  
  VLAN ID, 用于, 10  
  创建, 10-11, 16  
  创建 VNIC IP 接口, 16  
  带宽, 12  
  带有 VLAN ID, 16-17  
  更改底层链路, 26-28  
  配置, 18-20  
  迁移, 26-28  
  删除, 28-29

## VNIC (续)

  设置属性, 12  
  显示信息, 28  
  修改 MAC 地址, 26  
  修改 VLAN ID, 25-26  
  与区域结合使用, 20-22  
  指定 CPU 池资源, 40  
  指定给区域, 21  
  专用虚拟网络, 22-24

## 带

带宽  
  VNIC, 12  
  流的设置, 44  
  在流上, 13

## 丢

丢包, 55

## 动

动态环分组, 31-38

## 服

  服务级别协议 (service level agreement, SLA), 11  
  服务质量 (quality of service, QoS), 11

## 辅

  辅助客户机, 32, 36

## 管

  管理网络资源, 31



**环**

环, 传送和接收, 31-38

环分配, 请参见环分组

**环分组**

NIC 中的支持, 33-35

VLAN, 启用, 32

动态和静态, 31-38

分配步骤, 37-38

环使用和环分配, 36-37

接收和传送环, 33

解释环信息, 34, 35

数据链路属性, 32-33

显示信息, 32-33

**基**

基于软件的客户机, 31

基于硬件的客户机, 31

**监**

监视网络使用情况, 49

**接**

接口的专用 CPU, 42-43

**静**

静态环分组, 31-38

**扩**

扩展记帐功能, 57-61

IPQoS 的流记帐, 58

进程记帐, 58

链路和流的网络记帐, 58

任务记帐, 58

**流**

流, 12, 43-47

创建, 13, 43

带宽设置, 13, 44

设置属性, 13

属性, 12

显示信息, 44

流控制, 43-47

**内**

内部虚拟网络, 7-11

**迁**

迁移 vNIC, 26-28

**区****区域**

配置网络虚拟化, 18-20

虚拟网络配置, 20-22

指定 vNIC, 21

**属**

属性, 流, 12

**数****数据链路**

资源控制属性, 11, 12

数据链路, 资源控制属性, 12

数据中心整合, 10

**外**

外部虚拟网络, 7-11

## 网

- 网络记帐, 57-61
- 网络通道, 11
  - 软件通道, 49
  - 硬件通道, 49
- 网络统计数据, 52-55
  - 监视网络使用情况, 49
  - 聚合端口上的通信, 55
  - 流的资源使用情况, 56-57
  - 通道上的网络通信, 54-55
  - 通信历史信息, 58, 59-61
  - 网络通信设备的使用, 53-54
- 网络虚拟化, 7-11
  - etherstub, 8-9
  - VNIC, 8-9
  - 数据中心整合, 10
  - 虚拟交换机, 8-9
    - 用于实现的 dladm 子命令, 12
  - 组件, 8-9
- 网络资源管理, 31
  - CPU, 39
  - CPU 池, 39
  - 通过使用流, 12
  - 用于实现的 dladm 子命令, 12
  - 在链路上, 11

## 伪

伪以太网 NIC, 请参见 etherstub

## 虚

- 虚拟化和服务质量, 31
- 虚拟交换机, 8-9
- 虚拟网络
  - 另请参见 VNIC
  - etherstub, 15-16
  - 构建, 18-24
  - 配置 VNIC, 15-16
  - 配置区域, 18-20, 20-22
  - 外部和内部, 7-11
  - 与区域结合使用, 8
  - 专用, 7-11

## 虚拟网络 (续)

- 专用 IP 类型区域, 22
- 虚拟网卡, 请参见 VNIC

## 硬

- 硬件环, 31-38

## 中

- 中断, 42, 55

## 主

- 主客户机, 31, 36

## 专

- 专用虚拟网络, 7-11
  - 配置 etherstub, 22-24

## 资

- 资源控制, 11, 31