



# Sun Enterprise™ 10000 IDN 配置指南

---

Sun Microsystems, Inc.  
901 San Antonio Road  
Palo Alto, CA 94303-4900  
U.S.A. 650-960-1300

部件号: 806-6974-10  
2001年2月, 修订版 A

请将关于此文档的意见发送至: [docfeedback@sun.com](mailto:docfeedback@sun.com)

Copyright 2001 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, California 94303-4900 U.S.A. 版权所有。

本产品或文档受版权保护，且按照限制其使用、复制、分发和反编译的许可证进行分发。未经 Sun 及其许可证颁发机构的书面授权，不得以任何方式、任何形式复制本产品或本文档的任何部分。第三方软件，包括字体技术，由 Sun 供应商提供许可和版权。

本产品的某些部分从 Berkeley BSD 系统派生而来，经 University of California 许可授权。UNIX 是在美国和其它国家注册的商标，经 X/Open Company, Ltd. 独家许可授权。

Sun、Sun Microsystems、Sun 徽标、AnswerBook2、docs.sun.com、Sun Enterprise、SunFDDI、Sun StorEdge、OpenBoot 和 Solaris 是 Sun Microsystems, Inc. 在美国和其它国家的商标、注册商标或服务标记。所有的 SPARC 商标均按许可证使用，是 SPARC International, Inc. 在美国和其它国家的商标或注册商标。带有 SPARC 商标的产品均基于由 Sun Microsystems, Inc. 开发的体系结构。

OPEN LOOK 和 Sun™ 图形用户界面是 Sun Microsystems, Inc. 为其用户和许可证持有人开发的。Sun 承认 Xerox 在为计算机行业研究和开发可视或图形用户界面方面所作出的先行努力。Sun 以非独占方式从 Xerox 获得 Xerox 图形用户界面的许可证，该许可证涵盖实施 OPEN LOOK GUI 且遵守 Sun 的书面许可协议的许可证持有人。

本资料按“现有形式”提供，不承担明确或隐含的条件、陈述和保证，包括对特定目的的商业活动和适用性或非侵害性的任何隐含保证，除非这种不承担责任的声明是不合法的。

#### Sun Enterprise 10000 SSP 归属权：

本软件版权归 University of California 校务委员、Sun Microsystems, Inc. 和其它方共同所有。下列条款适用于与本软件相关的所有文件，除非在单独的文件里明确声明。

著作人允许不限用途免费使用、复制、修改、分发和许可本软件及其文档资料，但必须在所有副本中保留现有的版权声明，并在所有分发品中注明上述版权声明。一经授权使用，不再需要书面协议、许可证或专利权使用费。本软件的修改版权归修改者所有。如果修改者在所适用的每个文件的首页上明确附加了新的条款，所做的修改不受下列许可条款的制约。

在任何情况下，对于由使用本软件、文档资料及其衍生物而产生的，造成任何直接、间接、特殊、偶然或随之而来的损害，著作人和分销商概不负责，即使已告知著作人此类损害的可能性。

著作人和分销商特别声明，不承担任何保证，包括但不限于商业活动和特定目的的适用性和非侵害性的隐含保证。本软件按“现有形式”提供，著作人和分销商没有提供维护、支持、更新、优化或修改的义务。

这是 scotty，一种简单的 tcl 解释程序，它使用一些特殊命令来获得有关 TCP/IP 网络的信息。Copyright (c) 1993, 1994, 1995, J. Schoenwaelder, TU Braunschweig, Germany, Institute for Operating Systems and Computer Networks. 允许不限用途免费使用、复制、修改和分发本软件及其文档资料，但必须在所有副本中注明上述版权声明。University of Braunschweig 不负责本软件对任何用途的适用性。本资料按“现有形式”提供，不承担明确或隐含的保证。



请回收



Adobe PostScript

# 目录

---

- 前言 vii
- 阅读本书之前 vii
- 本书结构 vii
- 使用 UNIX 命令 viii
- 排印约定 viii
- Shell 提示符 ix
- 相关文档资料 ix
- 访问 Sun 联机文档资料 ix
- 订购 Sun 文档资料 x
- Sun 欢迎您提出宝贵意见 x
  
- 1. 域配置简介 1
  - 内存错误处理 1
  - 系统命令 1
    - snoop(1M) 命令 1
  
- 2. 配置域间网络 3
  - 域 IP 地址 3
  - 以太网和物理地址 4
  - 自动激活逻辑网络接口 4

- ▼ 启用自动激活网络接口 5
- 探测 IDN 接口 5
  - ▼ 探测 IDN 接口 5
  - ▼ 取消探测 IDN 接口 6
- 可调变量和参数 7
  - OpenBoot PROM 变量 7
    - ▼ 设置 OBP 变量 7
  - ndd(1M) 驱动程序参数 8
    - ▼ 设置 ndd(1M) 驱动程序参数 8
  - driver.conf(4) 参数 11
    - ▼ 设置 IDN 参数而不必重新引导 11
    - ▼ 设置 IDN 参数而需要重新引导 12
  - idn.conf(4) File 13
  - 必需匹配的参数 17
- 内核统计信息 18
  - kstat(3K) 统计信息 19
- 索引 23

# 表格列表

---

表 2-1	ndd(1M) 参数	9
表 2-2	IDN idn.conf(4) 文件参数	13
表 2-3	每个接口的 kstat(3K) 统计信息	20
表 2-4	kstat(3K) 全局统计信息	22



# 前言

---

本指南说明了 Sun Enterprise™ 10000 服务器的 InterDomain Network (IDN) 功能的配置。有关如何使用这些功能的信息，请参考第 ix 页上“相关文档资料”中列出的相应用户指南。

---

## 阅读本书之前

本指南适用于在 UNIX® 系统，尤其是在基于 Solaris™ 操作环境的系统方面有工作经验的 InterDomain Network 系统管理员。如果不具备这方面的知识，请首先阅读随本服务器一起提供的 AnswerBook2™ 格式的 Solaris 系统管理集中的所有书，并考虑进行 UNIX 系统管理培训。

此外，还要阅读并熟悉与服务器软件一起提供的 AnswerBook2 格式的《*TCP/IP and Data Communications Administration Guide*》。

---

## 本书结构

本书包括以下章节：

第一章包含对本指南的简介。

第二章说明如何更为高效和可靠地配置 IDN。

---

# 使用 UNIX 命令

本文档没有包括有关基本 UNIX 命令和过程（例如关闭系统、引导系统和配置系统）的信息。

有关此类信息，请参考以下文档资料：

- Solaris 2.x 软件环境（尤其是涉及 Solaris 系统管理的软件环境）的 AnswerBook2 联机文档资料
- 与系统一起收到的其它软件文档资料

---

# 排印约定

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。 % You have mail.
<b>AaBbCc123</b>	键入的内容，与计算机屏幕输出相区别	% <b>su</b> Password:
<i>AaBbCc123</i>	书名、新词或术语以及要强调的词	请阅读 《 <i>用户指南</i> 》的第六章。这些称为类选项。 要执行该操作， <i>必须</i> 是超级用户。
	命令行变量；将用实际名称或值来替代	要删除文件，键入 <code>rm 文件名</code> 。



---

## Shell 提示符

Shell	提示符
C shell	<i>machine_name%</i>
C shell superuser	<i>machine_name#</i>
Bourne shell and Korn shell	\$
Bourne shell and Korn shell superuser	#

---

## 相关文档资料

用途	书名	部件号
用户	《 <i>Sun Enterprise 10000 InterDomain Networks User Guide</i> 》	806-4131
	《 <i>TCP/IP and Data Communications Administration Guide</i> 》	805-4003
参考	《 <i>Sun Enterprise 10000 SSP 3.4 Reference Manual</i> 》	806-4871
	《 <i>Sun Enterprise 10000 IDN 错误消息</i> 》	806-6969

---

## 访问 Sun 联机文档资料

可以通过网站 [docs.sun.com](http://docs.sun.com)<sup>SM</sup> 访问 Web 上的 Sun 技术文档资料。可以在如下网址上浏览 [docs.sun.com](http://docs.sun.com) 文档或搜索特定的书目或主题：

<http://docs.sun.com>



---

**警告** – AnswerBook2 集的输出取决于在浏览器中选择的字体集。Sun Microsystems 建议您对于常规文字使用普通的 san-serif 字体，而对于屏幕文字则使用普通的等宽字体。

---

---

## 订购 Sun 文档资料

Fatbrain.com，一个 Internet 专业书店，储备了有关 Sun Microsystems, Inc. 的精选产品文档资料。

有关这些文档的列表及其订购方法，请访问 Fatbrain.com 上的 Sun Documentation Center，网址为：

<http://www.fatbrain.com/documentation/sun>

---

## Sun 欢迎您提出宝贵意见

我们致力于提高文档资料的质量，欢迎您提出宝贵意见和建议。您可以将意见通过电子邮件发送给我们，地址如下：

[docfeedback@sun.com](mailto:docfeedback@sun.com)

请在电子邮件的主题行中包括文档的部件号 (806-6974-10)。

# 域配置简介

---

本章包含对 《*Sun Enterprise 10000 IDN 配置指南*》的简介。有关如何安装和使用 InterDomain Network 的更多信息，请参考 SSP 3.4 AnswerBook2 集中的 《*Sun Enterprise 10000 InterDomain Networks User Guide*》。

---

## 内存错误处理

SMR 中的内存错误是由这些错误各自所在的域环境中遇到它们的处理器报告。如果从属域遇到 SMR 中的内存错误，则不向主域报告此错误。因此，主域就可以导出发生错误的内存，而忽略这些错误。

---

## 系统命令

本节介绍受 IDN 影响的系统命令并说明这些命令是如何受影响的。

### snoop(1M) 命令

snoop(1M) 命令只能支持有限的网络最大传输单元 (MTU) 值，而所有这些值远远小于 IDN 可以支持的大小。对于系统而言，IDN 驱动程序是作为标准以太网设备。

因此，如果要使用 `snoop(1M)` 命令捕获 IDN 数据传输，必须使用 `-s` 选项，并指定一个 1500 字节或更小的值，如下所示：

```
# snoop -d idn0 -s 1500
```

由于 IDN 的点到点特性，`snoop(1M)` 命令只能捕获来往于本地域之间的流量。

# 配置域间网络

---

本章包含有关逻辑网络接口的自动激活、影响 IDN 操作和性能的可调参数，以及设置可调参数的指导等信息。

---

## 域 IP 地址

任何标准传输控制协议 / 网际协议 (TCP/IP) 网络接口必须有一个指派的 IP 地址，以便域可以通过接口进行通讯。要建立 IDN 连接，一组域也必须有其指派的 IP 地址，而且该地址在要从域中访问的任何地址或子网中必须唯一。而且只有该 IDN 中的域才能看见这些地址。如果要将域用作外部主机与该域通过 IDN 连接到的其它域之间的路由器，在选择 IP 地址时必须要考虑 Sun Enterprise 10000 服务器所驻留网络的配置。通常，每个逻辑 IDN 接口均配置为单独的 IP 子网。IDN 软件不会使 IDN 成员域和 IP 地址产生任何关联，因此可以自由选择任何适合网络环境的 IP 地址。与指派的 IP 地址关联的主机名必须输入到 `/etc/hostname.idnX` 文件中，其中 `idnX` 代表指派了特定 IP 地址的逻辑 IDN 接口。这将使得该网络在引导该域之后自动启动。

注意，要启用 IDN 驱动程序并允许一个域成为 IDN 成员，必须至少创建一个 `/etc/hostname.idnX` 文件，以便在引导域时可以自动加载 IDN 驱动程序。只有当加载了 IDN 驱动程序后，SSP 才能将该域识别为 IDN 候选成员。

---

注意 – 缺省情况下，有八个可能的逻辑接口，从 idn0 到 idn7。通过使用 IDN 可调参数和 idn.conf(4) 文件，该值最大可以调整到 32（从 idn0 到 idn31）。只有具有相同活动 idnX 接口的域才能在同一 IDN 子网上相互通讯。

---

## 以太网和物理地址

ifconfig(1M) 命令允许您动态地更改网络接口的以太网地址或物理地址。但是，由于 IDN 的点对点特性，系统必须维护以太网地址中的识别信息以确定将数据包发向何处。因此，IDN 驱动程序不允许更改 IDN 接口的以太网地址或物理地址。这并不是问题，因为 IDN 是专用子网。即使安装的网络接口卡与 IDN 接口的物理地址相同，该假设仍然成立。

## 自动激活逻辑网络接口

IDN 逻辑网络接口（例如 idn0、idn1 等等）被视为与传统网络接口卡相同。尽管所有的 IDN 接口使用同一物理链接，但是在逻辑上这些接口是单独的网络接口；因此每个 IDN 接口都需要一个唯一的 /etc/hostname.idnX 文件以在引导该域时调用网络的自动探测。

/etc/hostname.idnX 文件只包含一项：与 IDN 接口关联的主机名或 IP 地址。如果 idn0 是 IDN 的逻辑网络接口，则 /etc/hostname.idnX 文件名应为 /etc/hostname.idn0，而且该文件中包含与该 IDN 接口关联的唯一的 hostname。

有关 /etc/hostname.idnX 文件内容的更多信息，请参考《*Sun Enterprise 10000 InterDomain Networks User Guide*》。此外，还可参考《*TCP/IP and Data Communications Administration Guide*》以获得有关 TCP/IP 配置文件的更多信息。

## ▼ 启用自动激活网络接口

执行以下步骤以创建 `/etc/hostname.idnX` 文件：

1. 在文本编辑器中打开一个新文件。
2. 键入 IDN 逻辑网络接口的名称或 IP 地址。
3. 将此文件保存为 `/etc/hostname.idnX`，其中 *X* 对应于要在引导时激活的 IDN 驱动程序实例。

如果域是 IDN 的成员，该域在引导时将自动与其它已引导的 IDN 成员链接起来，如 SSP 上的 `domain_status(1M)` 命令所示。Solaris `rc` 脚本与 `/etc/hostname.idnX` 文件一起使用，可以跨 IDN 启用逻辑网络接口。然后，该 IDN 将作为域间的标准 TCP/IP 网络。

---

注意 – IDN 的自动链接需要由 SSP 提供的服务。SSP 事件检测守护程序，`edd(1M)`，负责识别已引导域并执行 IDN 事件处理程序以进行实际链接。识别引导事件和 IDN 事件处理程序处理链接需要一段延迟时间，这取决于 SSP 上的负载大小。因此，到域的 IDN 链接完全运作之前该域可能已经引导完毕。该延迟时间不应超过几秒钟。

---

## 探测 IDN 接口

按照探测其它任何网络接口的方法探测 IDN 接口。为了方便起见，以下包含了有关信息。

## ▼ 探测 IDN 接口

对于链接到 IDN 的每个域中的每个 IDN 接口，都必须执行以下步骤。注意，在执行这些步骤之前，域不必链接到 IDN 上。

1. 探测每个域中的 IDN 接口。

```
# ifconfig idn0 plumb
```

在上例中，idn0 是基于使用 IPv4 的 IDN 接口名。有关正确使用 IPv6 的信息，请参考 IPv6 的文档资料。注意，在 Solaris 7 操作环境中不支持 IPv6。

*IP 地址* 定义为指派给各个主机的指定 IDN 接口的 IP 地址（有关更多信息，请参考《Sun Enterprise 10000 InterDomain Networks User Guide》中的“Domain IP Addresses”和 hosts(4) 手册页）。

2. 配置此 IDN 接口。

```
# ifconfig idn0 IP 地址 netmask 255.255.255.0 \  
broadcast IP 子网地址 up
```

上例中假定正在设置一个基本 IDN。如果要使用特定站点的网络掩码，用特定站点的值来替换网络掩码值。

## ▼ 取消探测 IDN 接口

不必取消正在从 IDN 中断开链接的域中的 IDN 接口探测。但是，要拆除整个 IDN，对于 IDN 中每个域的每个 IDN 接口，必须执行以下步骤。

1. 取消配置此 IDN 接口。

```
# ifconfig idn0 down
```

此步骤将拆除指定 IDN 接口的 TCP/IP 栈。

2. 取消探测此 IDN 接口。

```
# ifconfig idn0 unplumb
```



---

## 可调变量和参数

有几个变量和参数可以影响 IDN 的性能和资源使用。本节介绍如何设置这些变量和参数，包括它们的最小值、最大值以及缺省值。

### OpenBoot PROM 变量

OpenBoot™ PROM (OBP) 有一个与 IDN 相关的变量，必须修改它以启用 IDN，该变量为：共享内存区域 (SMR) 大小变量，`idn-smr-size`。此变量指定了 SMR 大小 (MB)。该值为零将禁用 IDN 网络。该值非零表示预留给 SMR 的内核空间空间数 (MB)。`idn-smr-size` 的缺省值为零 (0)。

SMR 越大，数据传送的可用缓冲区越大。但是，如果超出了某个阈值，较大的 SMR 值并不会带来其相应的优点。`idn-smr-size` 建议值为 32MB，该值对于大多数使用情况实用。其最大值为 96MB。

`idn-smr-size` 值只能在 OBP 提示符下进行设置。重新引导域之后，新值才能生效。但是，可以使用 `idn.conf` 文件中的 `idn_smr_size` 变量来减少 SMR 的实际大小。

---

注意 – IDN 中的所有域必须有相同的 `idn-smr-size` 值。如果有任何域的 `idn-smr-size` 值不正确，或如果要为整个 IDN 更改该值，必须重新引导受影响的域到 OBP 提示符下并重新设置该变量值。

---

#### ▼ 设置 OBP 变量

1. 在 `netcon(1M)` 窗口中，以超级用户身份登录到域。
2. 引导或终止域，使该域进入 OBP 提示符，并使用 `setenv` 命令设置此变量，如下所示：

```
<#0> ok setenv idn-smr-size 大小
```

3. 重新引导域。

#### 4. 成功地重新引导后，检查 OBP 设置。

```
<#0> ok cd /memory  
<#0> ok .properties
```

第二个命令将生成一个 OBP 变量及其相关设置的列表，如下所示：

```
idn-smr-size      00 00 00 20  
idn-smr-addr     00 00 00 0a 7d 3f 00 00 00 00 00 00 02 00 00 00  
dr-max-mem      00 00 9c 40  
reg             0000000a 00000000 00000000 80000000  
available       0000000a 7fff0000 00000000 00004000  
                0000000a 7fcd8000 00000000 00016000  
                0000000a 00000000 00000000 7189e000  
name            memory
```

如果已经正确分配了 SMR，idn-smr-addr 值应该不为零，它代表 SMR 的基本物理地址（例如，0xA7D3F0000）和大小的字节数（例如，0x2000000）。

## ndd(1M) 驱动程序参数

可以更改 ndd(1M) 驱动程序参数以调整系统的最佳性能和资源使用情况。本节解释了哪些参数可以更改、如何更改这些参数并列出了每个参数可用值的范围。

### ▼ 设置 ndd(1M) 驱动程序参数

#### 1. 读取当前的参数设置。

```
# ndd /dev/idn 参数
```

使用以下命令查看此 IDN 驱动程序支持的所有 ndd(1M) 参数的列表。

```
# ndd /dev/idn "?"
```

## 2. 更改该驱动程序参数。

```
# ndd -set /dev/idn 参数值
```

必须使用 `-set` 语法以修改本节提及的驱动程序参数。此外，除非另行说明，可以随时更改本节提及的所有驱动程序参数。

下表包括可以用 `ndd(1M)` 命令读取的参数名以及该参数的简要说明。有关 `ndd(1M)` 使用的更多信息，参见 `ndd(1M)` 手册页。

表 2-1 ndd(1M) 参数

名称	最小值	最大值	缺省值	说明
<code>idn_modunloadable</code>	0	1	0	此二进制标志表明是否可以卸载此 IDN 驱动程序（假定该驱动程序没有使用）。该值为 0 时标志关闭，为 1 时标志打开。该值可以随时进行更改。
<code>idn_slabpool</code>	n/a	n/a	n/a	如果域连接到 IDN 并且是 IDN 的主域，此参数将显示此 IDN 的阵列池，表明可用阵列的数目以及每个域分配了哪些阵列。该值为只读。
<code>idn_buffers</code>	n/a	n/a	n/a	显示域中未经要求的（即，无人认领的）SMR I/O 缓冲区数（就该域和其连接的其它域而言）。该值为只读。
<code>idn_mboxtbl</code>	n/a	n/a	n/a	显示分配给域的信箱表。如果该域不是 IDN 成员，将不会显示表。显示的信息包括信箱标题 <code>cookie</code> 、准备就绪和 / 或活动指针的值，以及指明各个通道服务器是否准备就绪和 / 或处于活动状态。该值为只读。
<code>idn_mboxtbl_all</code>	n/a	n/a	n/a	显示与 <code>idn_mboxtbl</code> 相同的信息；但是它显示的是整个 IDN 的相关信息。此参数只有在主域环境下执行才有效，因为该参数维护指向全局信箱区域的指针。

表 2-1 ndd(1M) 参数 (接上页)

名称	最小值	最大值	缺省值	说明
idn_mainmbox	n/a	n/a	n/a	包含了信箱管理结构 (该结构由域维护以将信箱发送其它 IDN 成员域, 或从中接收) 的详细信息。该值为只读。
idn_global	n/a	n/a	n/a	显示有关域的全局状态信息 (例如, 活动通道、与该域连接的域数, 以及 SMR 的物理地址)。它还显示 IDN 中每个域的连接状态概要。该值为只读。
idn_domain	n/a	n/a	n/a	显示有关域的特定状态信息 (例如, 未经要求的计数、令牌和未经要求的缓冲区数)。该值为只读。
idn_domain_all	n/a	n/a	n/a	显示类似于 idn_domain 的信息, 但是此信息包括该域连接的所有域的信息。该值为只读。
idn_bind_net	n/a	n/a	n/a	允许用户将特定的通道服务器 (接口) 联结到域中特定的处理器上, 因而可以更好地控制域中实际控制 IDN 数据接收的处理器。缺省情况下, 服务器未联结; 因此, 它们直接与普通线程竞争处理时间。自变量的格式为: 通道=cpuid。例如, 0=25 将把负责处理 idn0 接口上接收的数据的通道服务器联结到 cpuid 25。该值可以随时进行更改。

## driver.conf(4) 参数

IDN 允许使用 IDN 驱动程序的 driver.conf(4) 文件设置某些可调参数和 / 或配置参数。该文件的路径如下：

```
/platform/SUNW,Ultra-Enterprise-10000/kernel/drv/idn.conf
```

必须编辑 driver.conf(4) 文件以更改这些参数。其中的大多数参数都是全局参数。只有 bind\_cpu 参数是每个实例（接口）的参数。这些参数值将在使用 modload(1M) 命令加载此驱动程序后生效。

设置 IDN 参数的过程取决于域的当前状态。如果该域正在运行但没有链接到 IDN，则可以按照第 11 页“设置 IDN 参数而不必重新引导”中的指导设置这些 IDN 参数而不必重新引导域。如果该域没有运行或即将重新引导，可以按照第 12 页“设置 IDN 参数而需要重新引导”中的指导设置这些 IDN 参数。

### ▼ 设置 IDN 参数而不必重新引导

1. 确保该域没有链接到 IDN。
2. 在 netcon(1M) 窗口中，将目录更改为包含 idn.conf 文件的目录。

```
% cd /platform/SUNW,Ultra-Enterprise-10000/kernel/drv/
```

3. 编辑 idn.conf 文件，以使其反映出要使用的新值。
4. 取消探测所有的 IDN 网络接口。
5. 使用 ndd(1M) 命令以将 idn\_modunloadable 参数设置为正确值。

```
% ndd -set /dev/idn idn_modunloadable 1
```

6. 使用 modunload(1M) 命令卸载 IDN 驱动程序模块。

```
% modunload -i id
```

其中的 id 值必须与 IDN 模块 ID 编号的 ID 值对应。有关如何获得该模块 ID 编号的更多信息，请参考 modinfo(1M) 手册页。

7. 取消探测该 IDN 网络接口。

## ▼ 设置 IDN 参数而需要重新引导

1. 在 netcon(1M) 窗口中，将目录更改为包含 idn.conf 文件的目录。

```
% cd /platform/SUNW,Ultra-Enterprise-10000/kernel/drv/
```

2. 使用文本编辑器编辑该文件，以使其包含 IDN 的参数和值。

下例是一个 idn.conf 文件的示例。

```
name="idn" parent="pseudo" instance=0 bind_cpu=10;  
name="idn" parent="pseudo" instance=1;  
name="idn" parent="pseudo" instance=2 bind_cpu=35;  
idn_pil=4;  
idn_protocol_nservers=2;
```

对于所有必需的参数，必须编辑同一 IDN 中每个域的文件 idn.conf。对于其它所有参数，可以只编辑该域的文件 idn.conf。

每项可以占用多个行；但是必须用分号结束。在上例中，实例 0 的通道服务器 (idn0) 将联结到 CPU 10（假定该 CPU 在本系统中）。实例 1 的通道服务器 (idn1) 将不联结到系统中的任何 CPU 上，而实例 2 的通道服务器 (idn2) 将联结到 CPU 35（假定该 CPU 在本系统中）。

3. 重新引导域。

如果更改了需要匹配的参数设置，必须重新引导 IDN 中的每个域。如果更改了不需要匹配的要求设置，可以重新引导 IDN 中的一个域。参见第 17 页“必需匹配的参数”以获得必须匹配的参数列表。

## idn.conf(4) File

可以定义 `idn.conf(4)` 文件中某些参数的值，以便在使用 `modload(1M)` 命令加载 IDN 时设置这些参数。还可以将 IDN 实例添加到此文件中。编辑每个 IDN 实例（包含以下行，其中  $n$  等于实例数）的 `idn.conf(4)` 文件。

```
name="idn" parent="pseudo" instance= $n$ ;
```

**注意** – 可以在域链接到 IDN 时更改所有的 `idn.conf(4)` 文件参数；但只有在重新引导域后，更改的值才能生效。

下表包含参数名；参数的最小值、最大值和缺省值；以及指定这些参数的单元。



**警告** – 下表中的参数只能由受过培训的 IDN 用户使用。修改某些值可能会对 IDN 的性能产生负面影响。

表 2-2 IDN `idn.conf(4)` 文件参数

名称	最小值	最大值	缺省值	说明
<code>bind_cpu</code>	n/a	n/a	-1	指定哪一个 <code>cpuid</code> 在联机之后联结到各个通道服务器。此参数必须与特定的 CPU 实例关联。如果指定的 <code>cpuid</code> 不是域中有效的 CPU，该通道服务器将保留为没有联结。该值是 CPU 的 ID（-1 表示没有联结）。
<code>idn_awolmsg_interval</code>	0	3600	60	控制每个域控制台上显示 AWOL 消息的频率。该值是按秒计算。
<code>idn_checksum</code>	0	1	1	此二进制标志指明 SMR 信箱的校验和验证是否已打开。该值为 0 时标志关闭，值为 1 时标志打开。

表 2-2 IDN idn.conf(4) 文件参数 (接上页)

名称	最小值	最大值	缺省值	说明
idn_dmv_pending_max	8	512	128	控制单个处理器可以拥有的未经要求的 DMV 中断的最大数，这取决于 IDN 的驱动程序。它还说明了用于封装输入跨域中断数据的队列结构数。该值为数值。
idn_history	0	1	0	此二进制标志指明 IDN 是否应打开某些 IDN 事件的内部日志记录。其目的只是进行问题分析，以便为支持人员以后的调试收集信息。该值为 0 时标志关闭，为 1 时标志打开。
idn_hiwat	1024	1048576	262144	控制 IDN 流队列的高位线。该值是按字节计算。
idn_lowat	1	524288	1	控制 IDN 流队列的低位线。该值是按字节计算。
idn_max_nets	1	32	8	控制可以在 IDN 驱动程序上探测的网络通道或接口的最大数。该值以通用单位或数值计算。
idn_mbox_per_net	31	511	127	控制每个信箱表中的信箱项（通道和/或接口）的数量。该值必须是奇数。它以通用单位或数值计算。
idn_msgwait_cfg	10	300	40	控制等待 CFG（配置）消息响应的最短时间。该值是按秒计算。
idn_msgwait_cmd	10	300	40	控制等待 CMD（命令）消息（通常对于主域）响应的最短时间。该值是按秒计算。
idn_msgwait_con	10	300	20	控制等待 CON（连接）消息响应的最短时间。该值是按秒计算。
idn_msgwait_data	10	300	30	控制等待 DATA（断开连接）唤醒呼叫响应的最短时间。该值是按秒计算。



表 2-2 IDN idn.conf(4) 文件参数 (接上页)

名称	最小值	最大值	缺省值	说明
idn_msgwait_fin	10	300	40	控制等待 FIN (断开连接) 消息响应的最短时间。该值是按秒计算。
idn_msgwait_nego	10	300	20	控制等待 NEGO (协商) 消息响应的最短时间。该值是按秒计算。
idn_netsvr_spin_count	0	10000	500	控制通道服务器在放弃处理器之前巡回检测输入数据包的重复次数。该值以通用单位或数值计算。
idn_netsvr_wait_max	0	6000	1600	控制通道服务器在进入硬休眠之前将休眠的最大时钟报时信号数长。
idn_netsvr_wait_min	0	3000	40	控制通道服务器在没有找到输入数据包时将休眠的最初时钟报时信号数。该值以时钟报时信号计算 (100 个报时信号等于 1 秒)。
idn_netsvr_wait_shift	1	5	1	表示通道服务器在每次唤醒后因没有找到数据包而延长的休眠时间。该值为 1 将导致此时间增大为每次间隔时间的两倍。此休眠时间将一直增长, 直至等于由 idn_netsvr_wait_max 指定的最大值。该值以通用单位或数值计算。
idn_nwr_size	0	整个 SMR	整个 SMR	控制 SMR (此 SMR 用于以网络为基础的通讯) 的网络区域 (NWR) 部分的大小。该值是按 MB 计算。
idn_pil	1	9	8	控制软件中断的优先级, 并以此优先级处理跨域中断。该值为数值。
idn_protocol_nservers	1	16	4	控制委派至处理来自远程 IDN 成员域 IDN 连接消息的线程数。该值为数值。

表 2-2 IDN idn.conf(4) 文件参数 (接上页)

名称	最小值	最大值	缺省值	说明
idn_reclaim_max	0	128	0	控制域试图收回的未经要求、无人认领的最大缓冲区数。该值为 0 将导致域尽可能多地收回超出最小阈值 (idn_reclaim_min) 之外的缓冲区。该值是按缓冲区数计算。
idn_reclaim_min	1	128	5	控制未经要求 (无人认领) 的缓冲区数的阈值, 一旦超过该值, 域将试图收回缓冲区。该值是按缓冲区数计算。
idn_retryfreq_con	1	60	2	控制确认输入域到达 CON (连接) 阶段的两次重试之间的最短时间间隔。该值是按秒计算。
idn_retryfreq_fin	1	60	3	控制确认输出域到达 FIN (断开连接) 阶段的两次重试之间的最短时间间隔。该值是按秒计算。
idn_retryfreq_nego	1	60	2	控制启动 IDN 连接的两次重试之间的最短时间间隔。该值是按秒计算。
idn_sigbpil	1	9	3	控制软中断的优先级, 并以此优先级处理 SSP 信号块请求。该值为数值。
idn_slab_bufcount	4	1024	32	控制分配给每个阵列的缓冲区数。该值是按缓冲区数计算。
idn_slab_mintotal	2	16	8	控制主域维护的最小可用阵列数。如果可用阵列的总数少于此变量值时, 主域将请求从域返回未使用的阵列。该值是按阵列数计算。
idn_slab_prealloc	0	10	0	控制当域链接到 IDN 时预分配的阵列数。该值是按阵列数计算。
idn_smr_bufsize	512	524288	16384	控制转换为 IDN MTU 大小的 SMR I/O 缓冲区大小。该值按字节数计算, 并以 2 的次幂形式表示。

表 2-2 IDN idn.conf(4) 文件参数 (接上页)

名称	最小值	最大值	缺省值	说明
idn_smr_size	0	Entire SMR	0	SMR 大小受 OBP 变量 idn-smr-size 的值限制。SMR 的大小由变量 idn-smr-size 的最小值和参数 idn_smr_size 的最小值决定。如果将 idn-smr-size 设置为 0，OBP 变量会覆盖 idn.conf(4) 参数的值。该值是按 MB 计算。
idn_window_incr	0	32	8	控制由于每个附加活动通道和 / 或接口而增加的 idn_window_max 值。该值是按缓冲区数计算。
idn_window_max	8	256	64	控制未经要求的缓冲区数的基本阈值，一旦超过该值，域将停止将附加数据包发送到各个域。该值是按缓冲区数计算。

## 必需匹配的参数

同一 IDN 的所有域中的某些 IDN 参数必须相同。链接域的过程中交换配置信息时，在允许继续执行链接操作之前，每个域将验证接收到的信息是否与本地参数匹配。下表包含了在一个 IDN 中所有域的相应参数都必须相同的参数名。

- idn\_nwr\_size
- idn\_smr\_bufsize
- idn\_slab\_bufcount
- idn\_max\_nets
- idn\_mbox\_per\_net
- idn\_checksum

---

## 内核统计信息

IDN 驱动程序支持标准的 Solaris 内核统计信息机制，`kstat(3K)`。除了报告支持 `netstat(1M)` 所需的最小集外，IDN 驱动程序还报告可用于性能调整或配置管理的附加统计信息。获得这些统计信息的最简易的方法是使用标准的 `netstat(1M)` 或 `kstat(1M)` 命令行实用程序。

可以使用下例中的语法请求所有的统计信息。本例中是使用 `idn` 和 `idn0` 自变量获得的统计信息示例。

```
# netstat -k idn
idn:
curtime 2048474 reconfigs 0 reconfig_last 0 reaps 0 reap_last 0
links 1 link_last 2042885 unlinks 1 unlink_last 2045246 buf_fail 1
buf_fail_last 2042935 slab_fail 1 slab_fail_last 2042935
reap_count 0 dropped_intrs 0

# netstat -k idn0
idn0:
ipackets 3 ierrors 0 opackets 0 oerrors 0 collisions 0
rx_collisions 0 crc 0 buff 0 nolink 0 linkdown 0 inits 5 nocanput 0
allocbfail 0 notbufs 0 reclaim 0 smraddr 0 txmax 0 txfull 0 xdcall 3
sigsvr 10 mboxcrc 0 rbytes 238 obytes 238 multircv 0 multixmt 0
brdcstrcv 0 brdcstxmt 4 norcvbuf 0 noxmtbuf 0 ipackets64 3
opackets64 3 rbytes64 238 obytes64 238 fcs_errors 0
macxmt_errors 0 toolong_errors 0 macrcv_errors 0
```

可以请求单个名称或接口的统计信息，如下例所示（其中包括了作为逻辑网络接口的 idn0 和 idn1）。此例中的数值只是为了举例说明，收到的输出信息可能与其差别很大。

```
# netstat -k idn0 idn1

idn0:
ipackets 1386286 ierrors 0 opackets 1312137 oerrors 0 collisions 0
rx_collisions 0 crc 0 buff 0 nolink 0 linkdown 3561 inits 3
nocanput 131735 allocbfail 0 notbufs 0 reclaim 0 smraddr 0 txmax 0
txfull 0 xdcall 68783 sigsvr 63444 mboxcrc 0 rbytes 291362843
obytes 4225747350 multircv 0 multixmt 0 brdcstrcv 0 brdcstxmt 21
norcvbuf 131735 noxmtbuf 0 ipackets64 1386286 opackets64 1312131
rbytes64 13176264731 obytes64 12816667818 fcs_errors 0
macxmt_errors 16315 toolong_errors 0 macrcv_errors 0

idn1:
ipackets 189387 ierrors 0 opackets 136365 oerrors 0 collisions 0
rx_collisions 0 crc 0 buff 0 nolink 0 linkdown 0 inits 3
nocanput 54938 allocbfail 0 notbufs 0 reclaim 0 smraddr 0 txmax 0
txfull 0 xdcall 11788 sigsvr 453 mboxcrc 0 rbytes 1797429854
obytes 1226840176 multircv 0 multixmt 0 brdcstrcv 0 brdcstxmt 10
norcvbuf 54938 noxmtbuf 0 ipackets64 189387 opackets64 136364
rbytes64 1797429854 obytes64 1226840176 fcs_errors 0
macxmt_errors 0 toolong_errors 0 macrcv_errors 0
```

## kstat(3K) 统计信息

本节包含执行 netstat(1M) 命令（对于 IDN 驱动程序）时与该命令相关的 kstat(3K) 变量。注意，对于 idnX 项，将为所提供的每个网络接口报告此变量的各个实例。（在本表中，“无”表示不适用于 IDN。）

下表包括通过使用 netstat -k idn0 或 kstat -n idn0 可获得的每个实例统计信息的列表。

表 2-3 每个接口的 kstat(3K) 统计信息

统计信息	说明
allocbfail	IDN 驱动程序未能成功地将流缓冲区分配给输入消息的次数
brdcstrcv	接口收到的广播数据包的总数
brdcstxmt	接口传送的广播数据包的总数
buff	输入数据包大小超过 SMR I/O 缓冲区期望值的次数
collisions	无（传送冲突数），通常为零 (0)
crc	在收回缓冲区期间遇到或从远程域收到损坏的数据（报头）缓冲区的次数
fcs_errors	接收到的 IDN 数据包报头未通过 CRC 检验的数据包数
ierrors	输入错误的总数（例如，无法分配流缓冲区。信箱损坏或指定的缓冲区无效。）
inits	IDN 驱动程序的 init 函数的调用次数
ipackets	各个通道（网络接口）上的 IDN 驱动程序收到的数据包数
ipackets64	接口收到的数据包总数的 64 位计数器
linkdown	发现连接到指定域的现有 IDN 连接没有连接上的次数
macrcv_errors	所接收到的目标地址与接收接口地址不同的数据包数
macxmt_errors	由于内部 IDN 传送错误（例如连接中断），接口传送数据包失败 的次数
mboxcrc	域发送或接收信箱时遇到损坏的信箱报头的次数
multircv	接口收到的多点传送数据包的总数
multixmt	接口传送的多点传送数据包的总数
nocanput	在试图将数据推入协议栈时 IDN 驱动程序遇到完整流队列的次数
nolink	指定的目标域没有与本地域建立连接的次数
norcvbuf	无法分配缓冲区以接收输入数据包的次数
notbufs	域未能成功地将 SMR I/O 缓冲区分配给输出消息的次数
noxmtbuf	无法分配传送缓冲区以传送输出数据包的次数
obytes	接口传送的字节总数

表 2-3 每个接口的 kstat(3K) 统计信息 (接上页)

统计信息	说明
obytes64	接口传送的字节总数的 64 位计数器
oerrors	输出错误的总数 (例如发送信箱已损坏。无法分配 SMR I/O 缓冲区, 或数据包的报头已损坏。)
opackets	各个通道上的 IDN 驱动程序传送的数据包数
opackets64	接口传送的数据包总数的 64 位计数器
rbytes	接口收到的字节总数
rbytes64	接口收到的字节总数的 64 位计数器
reclaim	域试图收回输出缓冲区但是在该缓冲区中发现错误 (例如, 报头已损坏或遇到错误的 SMR 偏移量。) 的次数
rx_collisions	无 (接收冲突数), 通常为零 (0)
sigsvr	在收到跨域调用之后, 域必须通知通道服务器开始阅读其信箱的次数
smraddr	域遇到在 SMR 中指定的偏移量无效的 SMR I/O 缓冲区 (它只与接收域信箱中找到的输入缓冲区有关。) 的次数
toolong_errors	收到的大于预期 IDN MTU 大小的数据包数
txfull	当接收信箱已满时试图传送数据包的次数
txmax	当未经要求的数据包数超过 idn_window_emax 值时试图传送数据包的次数
xdcall	域必须执行跨域调用以通知接收器有关输入数据包的次数

下表包括通过使用 `netstat -k idn` 或 `kstat -n idn` 可获得的全局统计信息的列表。

表 2-4 kstat(3K) 全局统计信息

统计信息	说明
<code>buf_fail</code>	域未能成功地分配 SMR I/O 缓冲区的次数
<code>buf_fail_last</code>	在最近一次 SMR 缓冲区分配失败时 <code>lbolt</code> 的时间戳
<code>curtime</code>	在收集 <code>kstats</code> 以供其它时间戳参考而保存在全局 <code>kstats</code> 中时 <code>lbolt</code> 的瞬时状态
<code>dropped_intrs</code>	域因消息（协议）类型未知或 IDN 版本不合适而撤消的跨域调用（DMV 中断）总数
<code>link_last</code>	在最近一次出现链接、连接或请求时 <code>lbolt</code> 的时间戳
<code>links</code>	域参与的连接数（每个域连接计算为一个链接。）
<code>reap_count</code>	主域发出收回请求（即，要求归还）后，该域能够成功地收回的阵列总数（该值是根据域的周期进行累积的）
<code>reap_last</code>	最近一次执行收回操作时 <code>lbolt</code> 的时间戳
<code>reaps</code>	主域请求此域收回某些 SMR 阵列的次数
<code>reconfig_last</code>	最近一次执行重新配置时 <code>lbolt</code> 的时间戳
<code>reconfigs</code>	域参与重新配置的次数
<code>slab_fail</code>	域未能成功地从主域分配 SMR 阵列的次数
<code>slab_fail_last</code>	在最近一次 SMR 阵列分配失败时 <code>lbolt</code> 的时间戳
<code>unlink_last</code>	最近一次发出断开连接请求时 <code>lbolt</code> 的时间戳
<code>unlinks</code>	域参与断开连接操作的次数（每个域的断开连接计算为一个断开链接。）



# 索引

---

## 符号

edd(1M), 5

hosts(4), 6

### IDN

创建 IDN 网络的示例, 4

逻辑接口, 6

idn.conf(4), 13

文件参数, 13

参数匹配, 17

idn0 至 idn15, 6

### IDN 参数

重新引导, 12

不重新引导, 11

IPv6, 6

IP 地址, 域, 3

kstat(1M), 18

kstat(3K), 19

全局, 22

每个实例, 20

modload(1M), 13

ndd(1M) 驱动程序参数, 8

设置, 8

语法, 9

值, 9

netstat(1M), 18

OpenBoot PROM 变量, 7

缺省值, 7

设置, 7

大小, 7

## C

创建 IDN 网络, 示例, 4

创建 IDN 网络的示例, 4

## K

可调整的参数, 7

可调整的变量, 7

## L

逻辑接口, 3, 6

探测, 5

取消探测, 6

逻辑网络接口, 4, 5

自动激活, 5

## N

内存错误, 1

内核统计, 18

## **T**

调整最佳性能, 7

## **X**

性能调整, 7

系统命令, 1

## **Y**

域 IP 地址, 3

## **Z**

资源使用情况调整, 7