

在 Oracle® Solaris 11.2 中管理 SAN 设备和 多路径



文件号码 E53923-03
2014 年 12 月

版权所有 © 2009, 2014, 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

使用本文档	7
1 Solaris I/O 多路径概述	9
Solaris I/O 多路径概述	9
光纤通道软件功能	9
iSCSI 软件功能	10
SAS 软件功能	10
Solaris I/O 多路径功能	11
支持的设备标准	14
FC 设备配置概述	14
为 FC 设备配置多路径的注意事项	15
2 配置 Solaris I/O 多路径功能	17
配置多路径 I/O 功能	17
多路径注意事项	17
启用和禁用多路径	18
▼ 如何启用多路径	19
▼ 如何禁用多路径	20
▼ 如何确定多路径是启用还是禁用	21
基于每端口启用或禁用多路径	21
端口配置注意事项	21
配置第三方存储设备	23
第三方设备配置注意事项	23
配置第三方存储设备：新设备	24
配置第三方存储设备：禁用设备	25
显示设备名称更改	26
配置自动故障恢复	26
▼ 如何配置自动故障恢复	27
管理 Solaris I/O 多路径	27
显示多路径支持信息	27

为多路径支持配置自动故障恢复	35
3 配置光纤连接设备	43
FC 设备注意事项	43
添加 FC 设备	43
▼ 如何添加 FC 设备	44
在 SPARC 上配置光纤引导设备	44
光纤引导设备注意事项	44
4 配置 Solaris iSCSI 启动器	47
Oracle Solaris iSCSI 技术（概述）	47
确定 Oracle Solaris iSCSI 软件和硬件要求	48
配置 iSCSI 启动器任务	48
iSCSI 术语	49
建议的 iSCSI 配置做法	49
配置动态或静态目标搜索	50
▼ 如何配置 iSCSI 启动器	51
▼ 如何访问 iSCSI 磁盘	52
▼ 如何删除搜索到的目标设备	53
在基于 iSCSI 的存储网络中配置验证	54
▼ 如何配置对 iSCSI 启动器的 CHAP 验证	55
▼ 如何配置对 iSCSI 目标的 CHAP 验证	56
使用第三方 RADIUS 服务器简化 iSCSI 配置中的 CHAP 管理	57
设置 Oracle Solaris iSCSI 多路径设备	59
▼ 如何为目标启用多个 iSCSI 会话	60
监视 iSCSI 配置	62
▼ 如何显示 iSCSI 配置信息	63
修改 iSCSI 启动器和目标参数	64
调优 iSCSI 参数	65
▼ 如何修改 iSCSI 启动器和目标参数	67
iSCSI 配置问题的故障排除	70
从本地系统到 iSCSI 目标无连接	70
iSCSI 设备或磁盘在本地系统上不可用	71
在使用 iSNS 搜索方法时使用 LUN 屏蔽	72
一般 iSCSI 错误消息	72
5 配置虚拟光纤通道端口	77
什么是 NPIV？	77
NPIV 的限制	77

使用 NPIV 端口	78
▼ 如何创建 NPIV 端口	78
▼ 如何删除 NPIV 端口	78
▼ 如何显示 NPIV 端口状态	79
6 配置 FCoE 端口	81
什么是 FCoE ?	81
FCoE 的限制	81
配置 FCoE 端口	82
▼ 如何创建 FCoE 端口	82
▼ 如何删除 FCoE 端口	83
▼ 如何显示 FCoE 端口状态	83
▼ 如何强制 FCoE 端口重新初始化	84
配置 FCoE 硬件负载转移	85
7 配置 SAS 域	87
SAS 多路径注意事项	87
动态搜索 SAS 设备	87
配置 SAS 引导设备	88
8 配置 IPFC SAN 设备	89
IPFC 注意事项	89
确定光纤通道适配器端口实例	90
调用和配置 IPFC	92
▼ 如何手动启动网络接口	92
▼ 如何将主机配置为自动网络配置	93
9 在基于 x86 的系统上从光纤通道设备引导 Oracle Solaris OS	95
Oracle Solaris OS 安装要求	95
Oracle Solaris OS 安装概述	96
Oracle Solaris OS 安装过程	96
▼ 如何安装 Oracle Solaris OS	97
▼ 如何执行 DVD 或基于网络的 OS 安装	97
10 磁带设备的持久绑定	103
持久绑定概述	103
创建磁带链接	104
▼ 如何创建磁带设备链接	104

A 光纤连接的设备的手动配置	107
手动配置光纤连接的设备	107
▼ 如何手动配置光纤连接的设备	107
配置光纤设备节点	108
确保 LUN 级别信息可见	108
在未启用多路径的情况下配置设备节点	110
▼ 如何在不使用多路径的情况下手动配置 FC 设备	110
▼ 如何在没有使用多路径的情况下配置多个 FC 设备	111
在启用 Solaris 多路径的情况下配置设备节点	112
▼ 如何配置单个多路径 FC 设备	113
▼ 如何配置多个多路径 FC 设备	114
取消配置光纤设备	116
取消对光纤设备的配置	116
B 支持的 FC-HBA API	123
受支持的光纤通道 HBA API	123
C 多路径相关问题故障排除	125
在 stmsboot 期间系统引导失败或崩溃	125
▼ 如何在单用户模式中恢复引导故障	125
▼ 如何从系统崩溃中恢复	125
D 调优磁盘目标驱动程序属性	129
磁盘驱动程序的可调参数	129
用于调优磁盘驱动程序的名称:值对格式	130
用于调优磁盘驱动程序的位掩码格式	132
索引	135

使用本文档

- 概述 – 提供了面向 Oracle Solaris 操作系统的 Solaris I/O 多路径功能的概述，并介绍了如何配置 Solaris iSCSI 启动器、以太网光纤通道 (Fibre Channel over Ethernet, FCoE) 端口、存储区域网络 (Storage Area Network, SAN) 设备以及串行连接 SCSI (Serial-Attached SCSI, SAS) 域
- 目标读者 – 创建和维护光纤通道 (Fibre Channel, FC) 存储区域网 (Storage Area Network, SAN) 和串行连接 SCSI (Serial Attached SCSI, SAS) 域的系统管理员、存储管理员和网络管理员
- 必备知识 – SANS 和 SAS 域的管理和维护方面的专业知识

产品文档库

有关本产品的最新信息和已知问题均包含在文档库中，网址为：<http://www.oracle.com/pls/topic/lookup?ctx=E36784>。

获得 Oracle 支持

Oracle 客户可通过 My Oracle Support 获得电子支持。有关信息，请访问 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info>；如果您听力受损，请访问 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>。

反馈

可以在 <http://www.oracle.com/goto/docfeedback> 上提供有关本文档的反馈。

Solaris I/O 多路径概述

本章概述了 Solaris I/O 多路径功能（以前称为 StorageTek Traffic Manager 软件）。

本章包含以下主题：

- [“Solaris I/O 多路径概述” \[9\]](#)
- [“支持的设备标准” \[14\]](#)

Solaris I/O 多路径概述

Solaris I/O 多路径功能为运行 Oracle Solaris OS 的系统启用了多个访问路径。多路径功能通过使用多路径连接提高了存储设备的可用性。本节提供以下信息：

- [“光纤通道软件功能” \[9\]](#)
- [“SAS 软件功能” \[10\]](#)
- [“Solaris I/O 多路径功能” \[11\]](#)

Solaris I/O 多路径功能可识别 SAN 或 SAS 域上的存储设备。此软件使您可按环路、光纤或点对点模式连接光纤通道存储设备。此软件提供了一个可用于管理光纤通道、iSCSI 和 SAS 存储设备的公共界面。

有关如何在准备多路径时配置目标和启动器的信息，请参见 [《在 Oracle Solaris 11.2 中管理设备》](#) 中的第 8 章 [“使用 COMSTAR 配置存储设备”](#)。

光纤通道软件功能

Solaris I/O 多路径提供了以下重要功能：

- 动态存储器搜索 – 此软件可动态识别设备以及对设备配置所做的任何修改。此功能使您不必重新引导或手动更改配置文件中的信息，便可在系统上使用设备。
- 持久性设备命名 – 软件中配置的设备在重新引导或重新配置期间保留其设备命名。此策略只有一个例外情况，就是 `/dev/rmt` 中的磁带设备，这些设备不会发生更改，除非将其删除，然后在以后某个日期再重新生成。

- 光纤通道仲裁环路 (Fibre Channel Arbitrated Loop, FCAL) 支持 – 服务器上使用的 OpenBoot™ PROM (OBP) 命令可以访问 FCAL 连接的存储，以扫描 FC 环路。
- 结构网络引导 – Solaris OS 支持从结构网络设备和非结构网络光纤通道设备执行引导。带有光纤通道交换机的结构拓扑可提供更高的速度、更多的连接和端口隔离功能。
- FC-HBA 库 – 以前称为存储网络行业协会光纤通道主机总线适配器 (Storage Networking Industry Association Fibre Channel Host Bus Adapter, SNIA FC-HBA) 库，现在称为 FC-HBA 库。FC-HBA 库应用编程接口 (Application Programming Interface, API) 可用来管理 FC HBA，它为可用于收集 FC HBA 信息的其他应用程序 (如 Oracle StorEdge™ Enterprise Storage Manager) 提供了一个基于标准的接口。

有关常见 FC-HBA API 的更多信息，请参见 [libhbaapi\(3LIB\)](#)。有关 FC 规范的附加信息，请访问 <http://www.t11.org>。

- 光纤通道虚拟化 – N 端口 ID 虚拟化 (N Port ID Virtualization, NPIV) 是光纤通道标准的一个扩展，通过它一个光纤通道端口可在 SAN 上模拟成多个端口。这对虚拟化环境 (如 Oracle VM Server for SPARC 或适用于基于 x86 的系统的 Oracle VM Server 3.0) 很有用。
- 以太网光纤通道 (Fibre Channel over Ethernet, FCoE) – 现在可用的一个新 T11 标准，目的在于通过增强的以太网传输封装的光纤通道帧。Solaris FCoE 是一个经过专门设计能用于标准以太网控制器的软件实现。

iSCSI 软件功能

iSCSI 是 Internet SCSI (Small Computer System Interface, 小型计算机系统接口) 的首字母缩略词，是用于链接数据存储子系统的基于 Internet 协议 (Internet Protocol, IP) 的存储网络标准。通过在 IP 网络上传输 SCSI 命令，iSCSI 协议可用于通过网络访问块设备，就像这些设备连接至本地系统一样。

此功能意味着 Solaris 系统可以用作 iSCSI 服务器 (*target*) 或客户机 (*initiator*)。设置 Solaris iSCSI 目标的优点是，您可以拥有无需其他 FC HBA 就可连接至客户机的现有光纤通道设备。此外，具有专用阵列的系统现在可以与 ZFS 或 UFS 文件系统共享已复制的存储。

有关详细信息，请参见第 4 章 [配置 Solaris iSCSI 启动器](#)。

SAS 软件功能

- 动态存储器搜索 – Oracle Solaris OS 多路径软件可自动识别设备以及对设备配置所做的任何修改。此功能使您不必重新引导或手动更改配置文件中的信息，便可在系统上使用设备。
- 持久性设备命名 – Solaris OS 多路径软件中配置的动态存储器搜索设备，在重新引导或重新配置期间保留其设备命名。

Solaris I/O 多路径功能

缺省情况下，基于 x86 的平台将启用 Solaris I/O 多路径功能，此功能对于运行 Oracle Solaris OS 的基于 SPARC 的系统则是可选的。此软件包含下列功能：

- 路径管理 – Solaris I/O 多路径功能可动态管理指向 OS 所支持的任何存储设备的路径。当路径被置于联机状态或从服务中删除时，系统会自动添加或删除指向某个设备的路径。即使在启用了多路径功能的情况下，您也可以添加更多的控制器以增加带宽和 RAS，而不必更改设备名称或修改应用程序。Oracle 存储产品不要求管理配置文件或将数据库保持为最新状态。对于由供应商而非 Oracle 生产的存储器，请联系供应商，咨询启用支持的方法并确保此产品适合使用 Solaris I/O 多路径功能。
- 单个设备实例 – Solaris I/O 多路径功能完全与 Oracle Solaris OS 集成在一起。多路径设备显示为单个设备实例，而不是每个路径一个设备或设备链接。此功能使用诸如 `format` 命令的实用程序或卷管理产品来缩减管理复杂存储体系结构的成本，一个存储设备您只能看到一个表示，而不是每个路径对应一个单独的设备。
- 故障转移支持 – 实现更高级别的 RAS 需要具有到存储设备的冗余主机连接。Solaris I/O 多路径功能可管理存储路径故障，同时通过可用的辅助路径保持主机 I/O 的连通性。

您可以通过使用下列命令确定设备是否支持故障转移：

```
# mpathadm show mpath-support libmpscsi_vhci.so
mpath-support: libmpscsi_vhci.so
Vendor: Sun Microsystems
Driver Name: scsi_vhci
Default Load Balance: round-robin
Supported Load Balance Types:
    round-robin
    logical-block
Allows To Activate Target Port Group Access: yes
Allows Path Override: no
Supported Auto Failback Config: 1
Auto Failback: on
Failback Polling Rate (current/max): 0/0
Supported Auto Probing Config: 0
Auto Probing: NA
Probing Polling Rate (current/max): NA/NA
Supported Devices:
.
.
.
```

- 对称/非对称设备支持 – 支持下列磁盘存储设备：
 - 所有 Oracle 磁盘存储产品，无论是对称还是非对称的。
 - 所有符合 T10/T11 标准的第三方对称磁盘设备。
 - 许多第三方非对称磁盘阵列。

- 对于支持此 T10 标准的非对称设备，增加了 T10 非对称逻辑单元访问 (Asymmetric Logical Unit Access, ALUA) 支持。请联系您的存储器供应商，以确定您的设备是否受支持。

如果磁盘存储阵列使用 `f_asym_lsi` 故障转移模块，您需要手动更改 `ddi-forceload` 部分的序列，方法是将 `f_asym_lsi` 移到 `scsi_vhci.conf` 文件的结尾，类似于如下方式：

```
ddi-forceload =
    "misc/scsi_vhci/scsi_vhci_f_asym_sun",
    "misc/scsi_vhci/scsi_vhci_f_asym_emc",
    "misc/scsi_vhci/scsi_vhci_f_sym_emc",
    "misc/scsi_vhci/scsi_vhci_f_sym_hds",
    "misc/scsi_vhci/scsi_vhci_f_sym",
#    "misc/scsi_vhci/scsi_vhci_f_tpgs_tape",
#    "misc/scsi_vhci/scsi_vhci_f_tape",
    "misc/scsi_vhci/scsi_vhci_f_tpgs",
    "misc/scsi_vhci/scsi_vhci_f_asym_lsi";
```

- I/O 负载平衡 – 除了提供简单的故障转移支持，Solaris 多路径功能还可以使用指向存储设备的任何活动路径来发送和接收 I/O。由于可通过多个主机连接路由 I/O，可通过增加主机控制器来增加带宽。此软件使用循环（共享）负载平衡算法，按此算法，各个 I/O 请求按顺序逐一路由至活动的主机控制器。
- 队列深度 – SCSI 存储阵列以逻辑单元号 (Logical Unit Number, LUN) 形式向系统提供存储器。LUN 具有一个有限的可用资源集，如可以存储的数据量，以及设备或 LUN 一次可以处理的活动命令数量。在设备阻止后来的 I/O 之前可以发出的活动命令数量称为队列深度。启用了 Solaris I/O 多路径功能时，会为每个 LUN 创建一个队列，而不考虑 LUN 可能拥有多少个指向主机的不同或单独路径。使用此功能磁盘驱动程序可为 LUN 维护一个队列并使之平衡，有效地管理了队列深度。没有其他可用于 Oracle Solaris OS 的多路径软件具有此功能。
- `stmsboot` 命令 – Oracle Solaris OS 包含 `stmsboot` 命令，该命令允许您在完成 OS 安装后在引导设备上启用或禁用 Solaris 多路径功能。此命令既适用于基于 SPARC 的系统和也适用于基于 x86 的系统，并提供了 SAS 多路径支持。
缺省情况下，基于 SPARC 或 x86 的系统不会在 Oracle Solaris OS 中启用 SAS 多路径，所以如果希望启用多路径，需要在安装后使用 `stmsboot` 命令。
缺省情况下，基于 SPARC 的系统不会为 FC 设备启用多路径，所以需要在安装后使用 `stmsboot` 命令。
缺省情况下，基于 X86 的系统会为 FC 设备启用多路径，所以不要求在安装后使用 `stmsboot` 命令。
- 动态重新配置 – Solaris I/O 多路径功能支持动态重新配置 (Dynamic Reconfiguration, DR) 操作。
- 多路径设备名称 – 在系统上安装了 Oracle Solaris OS 并启用了 Solaris I/O 多路径后，多路径设备名称以 `c0` 开头。例如：

```
# mpathadm list lu
/dev/rdisk/c0t600A0B800026D63A0000A4994E2342D4d0s2
```

```

Total Path Count: 4
Operational Path Count: 4
/dev/rdisk/c0t600A0B800029065C00007CF54E234013d0s2
Total Path Count: 4
Operational Path Count: 4
/dev/rdisk/c0t600A0B800026D63A0000A4984E234298d0s2
Total Path Count: 4
Operational Path Count: 4
/dev/rdisk/c0t600A0B800029065C00007CF44E233FCFd0s2
Total Path Count: 4
Operational Path Count: 4
/dev/rdisk/c0t600A0B800026D63A0000A4974E23424Ed0s2
Total Path Count: 4
Operational Path Count: 4
/dev/rdisk/c0t600A0B800029065C00007CF34E233F89d0s2
Total Path Count: 4
Operational Path Count: 4
/dev/rdisk/c0t600A0B800026D63A0000A4964E234212d0s2
Total Path Count: 4
Operational Path Count: 4
/dev/rdisk/c0t600A0B800026D63A000099B94DE2DB34d0s2
.
.
.

```

- 显示 MPxIO 路径信息 – 更新了 prtconf 和 fmdump 命令以提供 MPxIO 路径信息。例如，下列输入就是来自带有多路径设备的系统。

```

# prtconf -v | grep path | more
Paths from multipath bus adapters:
  name='path-class' type=string items=1
  name='path-class' type=string items=1
  name='path-class' type=string items=1
  name='path-class' type=string items=1
    dev_path=/scsi_vhci/disk@g600a0b800026d63a0000a4994e2342d4:a
    dev_path=/scsi_vhci/disk@g600a0b800026d63a0000a4994e2342d4:a,raw
    dev_path=/scsi_vhci/disk@g600a0b800026d63a0000a4994e2342d4:b
    dev_path=/scsi_vhci/disk@g600a0b800026d63a0000a4994e2342d4:b,raw
    dev_path=/scsi_vhci/disk@g600a0b800026d63a0000a4994e2342d4:c
    dev_path=/scsi_vhci/disk@g600a0b800026d63a0000a4994e2342d4:c,raw
    dev_path=/scsi_vhci/disk@g600a0b800026d63a0000a4994e2342d4:d
.
.
.

```

diskinfo 命令可显示机箱、插口和磁盘插卡信息或托架插口的磁盘插卡，也可以用于显示多路径设备名称。例如：

```
# diskinfo -O occupant-compdev
c0t50015179594B6F11d0
c0t5000C500335F95E3d0
c0t5000C500335F907Fd0
c0t5000C500335BD117d0
c0t5000C500335DC60Fd0
c0t5000C500335E106Bd0
c0t5000C500335BA8C3d0
c0t5000C500335FC3E7d0
```

- 设备驱动程序配置 – 驱动程序的定制在 /etc/driver/drv 目录中进行，而不像以前发行版那样在 /kernel/drv 目录中进行。此改进意味着在系统升级时，您的驱动程序定制不会被覆盖。在升级期间保留 /etc/driver/drv 目录中的文件。对 fp.conf、mpt.conf 或 scsi_vhci.conf 的任何修改均应在 /etc/driver/drv 目录中进行。

支持的设备标准

Solaris I/O 多路径功能基于与设备和设备管理进行通信，确保与其他基于标准设备和软件的互操作性的开放标准。Solaris I/O 多路径功能支持下列标准：

- ANSI 标准：信息技术 – SNIA 多路径管理 API 规范 (ANSI INCITS 412-2006)
- T10 标准，包括 SCSI-3、SAM、FCP、SPC 和 SBC
- T11.3 FC 标准，包括 FC-PH、FC-AL、FC-LS 和 FC-GS
- T11.5 存储管理标准，包括 FC-HBA
- IETF 标准，包括 RFC 2625
- 串行连接 SCSI-2 (SAS2)

FC 设备配置概述

本节是对如何为 FC 设备配置 Solaris I/O 多路径功能的总体概述。

下表列出了配置任务、每项任务的说明以及可从其中找到相关过程的参考信息。

表 1-1 配置多路径 FC 设备的任务

配置任务	任务描述	参考信息
启用多路径功能	按如下所述启用多路径： ■ 缺省情况下，为 x86/x64 平台上的 FC 设备启用 ■ 在 SPARC 平台上通过手动配置启用	第 2 章 配置 Solaris I/O 多路径功能。

配置任务	任务描述	参考信息
	■ 为 SAS 设备通过手动配置启用	
配置 FC 设备	在 Oracle Solaris OS 中，FCAL、光纤和点对点连接的设备可供主机使用。	第 3 章 配置光纤连接设备
配置虚拟 FC 端口	可配置 N 端口 ID 虚拟化 (N Port ID Virtualization, NPIV) 端口，专门用于虚拟化环境。	第 5 章 配置虚拟光纤通道端口
设置 FC 引导设备	Solaris I/O 多路径功能支持 Solaris 服务器从光纤设备引导。	“FC 设备注意事项” [43]
配置 IPFC SAN 设备	在 SAN 中，可配置 IPFC 设备的主机识别和 IP over FC 的实现。IPFC 的配置取决于 FC 适配器端口的光纤端口 (Fabric Port, FP) 驱动程序实例。	第 8 章 配置 IPFC SAN 设备
配置 SAS 设备	在 Oracle Solaris OS 中，支持 SAS 设备的多路径。	第 7 章 配置 SAS 域

为 FC 设备配置多路径的注意事项

配置 Solaris I/O 多路径功能之前，请注意以下事项：

- 按照供应商特定文档为存储和交换机配置端口和区域。
- LUN 屏蔽支持特定 LUN 显示给特定主机。参见用于介绍屏蔽的供应商特定存储文档。
- 需要为 SAN 上的主机和设备禁用电源管理。有关电源管理的更多信息，请参见 [poweradm\(1M\)](#)。
- STMS 引导实用程序包括在管理 SAN 引导进程的 Solaris I/O 多路径功能中。发出 `stmsboot` 命令将自动更新 `/etc/vfstab` 文件和转储配置，以在启用或禁用多路径软件时反映设备名称更改。请注意，缺省情况下，该软件对于基于 SPARC 的系统上的设备处于禁用状态，对于基于 x86 的系统上的设备处于启用状态。
- 与光纤相连的设备已进行配置，在安装期间和进行引导时可自动供主机使用。

注 - 如果您正在执行升级并希望任何 FC 设备在升级之后不可用，必须使用 `cfgadm -c unconfigure` 命令手动取消这些设备的配置。不过，要使这些设备对于系统永久不可用，可能需要考虑使用交换机区域划分或 LUN 屏蔽。由 `cfgadm -c unconfigure` 进行的更改在重新引导后将不再存在，除非对 FC 设备的手动配置已经启用。要了解如何在引导或安装期间禁用 FC 设备发现，请参阅[附录 A, 光纤连接的设备的手动配置](#)。

配置 Solaris I/O 多路径功能

本章介绍如何在 Oracle Solaris OS 中为 FC 设备配置 Solaris I/O 多路径功能。还提供了在基于 SPARC 的系统上、基于 x86 的系统上、基于每端口和在第三方存储设备上启用或禁用多路径功能时的注意事项。

本章包含以下主题：

- “配置多路径 I/O 功能” [17]
- “多路径注意事项” [17]
- “启用和禁用多路径” [18]
- “基于每端口启用或禁用多路径” [21]
- “配置第三方存储设备” [23]
- “配置自动故障恢复” [26]

配置多路径 I/O 功能

可以为 FC 设备配置多路径功能以控制所有受支持的 FC HBA。缺省情况下，在基于 SPARC 的系统上禁用 FC 设备的多路径功能，而在基于 x86 的系统上启用这一功能。对多路径功能的配置取决于您打算如何使用系统。

注 - 并行 SCSI 设备不能使用多路径功能，但 FC、SAS 和 iSCSI 设备可以使用这一功能。磁带机和磁带库也支持多路径。

多路径注意事项

在更改多路径配置之前，请注意以下事项。然后按照随后几节中所述的针对您的计算机体系结构（SPARC 或 x86）的说明进行操作。某些设备需要进行适当配置才能使用多路径软件。有关特定于您设备的配置的详细信息，请参阅您的存储阵列文档。

- 特定于设备及设备名称更改注意事项：

在 `/dev` 和 `/devices` 树中，多路径设备会使用可表明其处于多路径控制之下的新名称。因此，当某个设备处于多路径控制之下时，它会有一个不同于其原始名称的名称。

禁用多路径时的设备名称：

```
/dev/dsk/clt1d0s0
```

启用多路径时的设备名称：

```
/dev/dsk/c0t60003BA27D5170003E5D2A7A0007F3D2d0s0
```

因此，将多路径配置从禁用更改为启用时，必须将直接使用设备名称的应用程序配置为使用新名称；反之亦然。

- 对 `/etc/vfstab` 条目和转储配置的更新：

系统的 `/etc/vfstab` 文件和转储配置也包含对设备名称的引用。在基于 SPARC 和基于 x86 的系统上，`stmsboot` 命令都会自动用新的设备名称更新 `/etc/vfstab` 文件转储配置。如果您的依赖于应用程序的文件系统未在 `/etc/vfstab` 文件中列出，可使用 `stmsboot` 命令确定新旧设备路径之间的映射。



注意 - 如果您已经运行 `devfsadm -C` 或已经执行重新配置引导，则旧的设备路径将不再存在，`stmsboot -L` 命令将无法提供此信息。

启用和禁用多路径

可使用 `stmsboot` 命令为光纤通道 (Fibre Channel, FC) 和 SAS 设备启用或禁用多路径。 `stmsboot` 命令将更新 `/etc/vfstab` 文件和转储配置，以在下次重新引导时反映设备名称更改。您无需手动编辑 `fp.conf` 或 `mpt.conf` 文件。

以下注意事项适用于 `stmsboot -e`、`-d` 和 `-u` 选项：

- 运行 `stmsboot` 命令之后应立即重新引导。
- 因为 `stmsboot` 命令会重新引导计算机以完成操作，请使用 `eeeprom` 以确保系统从当前引导设备进行引导。
- `stmsboot` 命令会在修改原始 `/kernel/drv/fp.conf`、`/kernel/drv/mpt.conf`，和 `/etc/vfstab` 文件之前，保存它们的副本。如果您在使用 `stmsboot` 命令时遇到意外问题，请参见[附录 C, 多路径相关问题故障排除](#)。

注 - 在先前的 Oracle Solaris 发行版中，`stmsboot` 命令仅用于为基于 SPARC 的主机在引导设备上启用或禁用多路径。在当前的 Oracle Solaris 发行版中，该命令用于在串行连接的 SCSI 设备以及 FC 设备上启用或禁用多路径。

▼ 如何启用多路径

在基于 SPARC 或 x86 的系统上，执行以下步骤在所有具有多路径功能的设备上启用多路径。如果您只想在特定的 FC 或 SAS HBA 端口上启用多路径，请参阅[“基于每端口启用或禁用多路径” \[21\]](#)。

多路径软件会自动识别 Oracle 支持的设备。如果您想要在第三方设备上启用多路径，请将 `/kernel/drv/scsi_vhci.conf` 文件复制到 `/etc/driver/drv`，然后如[“多路径注意事项” \[17\]](#)中的“设备名称更改注意事项”所述添加条目。

1. 成为管理员。
2. 确认多路径软件包是否已安装。

```
# pkg info system/storage/multipath-utilities
      Name: system/storage/multipath-utilities
      Summary: Solaris Multipathing CLI
      Description: Path Administration Utility for a Solaris Multipathing device
      Category: Applications/System Utilities
      State: Installed
      Publisher: solaris
      Version: 0.5.11
      Build Release: 5.11
      Branch: 0.175.0.0.0.0.0
      Packaging Date: Tue Sep 27 01:40:01 2011
      Size: 77.29 kB
      FMRI: pkg://solaris/system/storage/multipath-utilities@
0.5.11,5.11-0.175.0.0.0.0.0:20110927T014001Z
```

如果未安装，请进行安装。

```
# pkg install system/storage/multipath-utilities
```

3. 启用设备多路径。

```
# stmsboot -e
WARNING: stmsboot operates on each supported multipath-capable controller
detected in a host. In your system, these controllers are

/devices/pci@780/pci@0/pci@8/SUNW,qlc@0/fp@0,0
/devices/pci@780/pci@0/pci@8/SUNW,qlc@0,1/fp@0,0
/devices/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@1
/devices/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@1
/devices/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2
/devices/pci@7c0/pci@0/pci@9/LSILogic,sas@0
/devices/pci@7c0/pci@0/pci@9/LSILogic,sas@0

If you do NOT wish to operate on these controllers, please quit stmsboot
and re-invoke with -D { fp | mpt } to specify which controllers you wish
to modify your multipathing configuration for.

Do you wish to continue? [y/n] (default: y) y
```

```
Checking mpxio status for driver fp
Checking mpxio status for driver mpt
WARNING: This operation will require a reboot.
Do you want to continue ? [y/n] (default: y) y
The changes will come into effect after rebooting the system.
Reboot the system now ? [y/n] (default: y) y
```

注 - 在重新引导期间，/etc/vfstab 和转储配置将更新，以反映设备名称更改。

4. (可选) 重新引导之后，如有必要，按“[多路径注意事项](#)” [17]中的“设备名称更改注意事项”所述，将您的应用程序配置为使用新的设备名称。

▼ 如何禁用多路径

在基于 SPARC 或 x86 的系统上，执行以下步骤在所有具有多路径功能的设备上禁用多路径。如果您只想在特定的 FC 或 SAS HBA 端口上禁用多路径，请参阅[“基于每端口启用或禁用多路径”](#) [21]。

1. 禁用设备多路径。

```
# stmsboot -d
WARNING: stmsboot operates on each supported multipath-capable controller
        detected in a host. In your system, these controllers are

/devices/pci@780/pci@0/pci@8/SUNW,qlc@0/fp@0,0
/devices/pci@780/pci@0/pci@8/SUNW,qlc@0,1/fp@0,0
/devices/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@1
/devices/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@1
/devices/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2
/devices/pci@7c0/pci@0/pci@9/LSILogic,sas@0
/devices/pci@7c0/pci@0/pci@9/LSILogic,sas@0

If you do NOT wish to operate on these controllers, please quit stmsboot
and re-invoke with -D { fp | mpt } to specify which controllers you wish
to modify your multipathing configuration for.

Do you wish to continue? [y/n] (default: y) y
Checking mpxio status for driver fp
Checking mpxio status for driver mpt
WARNING: This operation will require a reboot.
Do you want to continue ? [y/n] (default: y) y
The changes will come into effect after rebooting the system.
Reboot the system now ? [y/n] (default: y) y
```

注 - 在重新引导期间，/etc/vfstab 和转储配置将更新，以反映设备名称更改。

2. (可选) 重新引导之后，如有必要，按“[多路径注意事项](#)” [17]中的“设备名称更改注意事项”所述，将您的应用程序配置为使用新的设备名称。

▼ 如何确定多路径是启用还是禁用

1. 成为管理员。
2. 确定系统上的多路径设备信息是否可用。
因为以下命令未返回多路径设备信息，所以多路径在该系统中已禁用。

```
# prtconf -vc /devices/scsi_vhci |grep dev_link.*s2
```

因为以下命令返回了多路径设备信息，所以多路径在该系统中已启用。

```
# prtconf -vc /devices/scsi_vhci |grep dev_link.*s2
dev_link=/dev/dsk/c0t5000C500335DC60Fd0s2
dev_link=/dev/rdisk/c0t5000C500335DC60Fd0s2
dev_link=/dev/dsk/c0t5000C500335E106Bd0s2
dev_link=/dev/rdisk/c0t5000C500335E106Bd0s2
dev_link=/dev/dsk/c0t5000C500335BA8C3d0s2
dev_link=/dev/rdisk/c0t5000C500335BA8C3d0s2
dev_link=/dev/dsk/c0t5000C500335FC3E7d0s2
dev_link=/dev/rdisk/c0t5000C500335FC3E7d0s2
dev_link=/dev/dsk/c0t50015179594B6F11d0s2
dev_link=/dev/rdisk/c0t50015179594B6F11d0s2
dev_link=/dev/dsk/c0t5000C500335F95E3d0s2
dev_link=/dev/rdisk/c0t5000C500335F95E3d0s2
dev_link=/dev/dsk/c0t5000C500335F907Fd0s2
dev_link=/dev/rdisk/c0t5000C500335F907Fd0s2
dev_link=/dev/dsk/c0t5000C500335BD117d0s2
dev_link=/dev/rdisk/c0t5000C500335BD117d0s2
```

基于每端口启用或禁用多路径

可在特定的光纤通道主机总线适配器 (Host Bus Adapter, HBA) 控制器端口上启用或禁用多路径。如果在特定的 HBA 控制器端口上启用多路径，则与该控制器端口相连的所有受支持的设备都将启用多路径操作。

端口配置注意事项

开始按端口配置软件之前，请注意以下事项：

- FC 全局和每端口多路径设置已在 `/kernel/drv/fp.conf` 文件中指定。
每端口多路径设置的优先级高于全局设置。因此，如果已启用全局多路径，但某个特定端口禁用了多路径，则该端口在多路径配置中将不可用。相反，即使全局多路径已禁用，如果特定的端口在相应的 `driver.conf` 文件中列出，也可启用这些端口的多路径功能。

- 负载均衡由 `/kernel/drv/scsi_vhci.conf` 文件中的全局负载均衡属性控制，而不是按端口控制。
- 如果某设备与主机之间有多条路径，则连接设备的所有路径必须均配置为启用或禁用多路径。
- 按端口配置多路径支持多路径软件与其他多路径解决方案（如 Symantec (VERITAS) 动态多路径 (Dynamic Multipathing, DMP) 或 EMC PowerPath）共存。不过，不应在多路径软件与其他多路径解决方案之间共享设备和路径。

▼ 如何按端口配置多路径

以下过程同时适用于基于 SPARC 的系统和基于 x86 的系统。

根据您希望多路径软件所控制的端口数，可全局或针对指定端口来启用或禁用多路径。

1. 成为管理员。
2. 确定您希望多路径软件控制的 HBA 控制器端口。

要列出可用设备，请对 `/dev/cfg` 目录执行 `ls -l` 命令。以下示例显示 `ls -l` 命令输出。

```
# ls -l
lrwxrwxrwx 1 root root 50 Jan 29 21:33 c0 ->
    ../../devices/pci@7c0/pci@0/pci@1/pci@0/ide@8:scsi
lrwxrwxrwx 1 root root 61 Jan 29 21:33 c1 ->
    ../../devices/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@1:scsi
lrwxrwxrwx 1 root root 61 Jan 29 21:33 c2 ->
    ../../devices/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2:scsi
lrwxrwxrwx 1 root root 53 Jan 29 21:33 c3 ->
    ../../devices/pci@7c0/pci@0/pci@9/LSILogic,sas@0:scsi
lrwxrwxrwx 1 root root 54 Apr 16 20:28 c5 ->
    ../../devices/pci@780/pci@0/pci@8/SUNW,qlc@0/fp@0,0:fc
lrwxrwxrwx 1 root root 56 Apr 16 20:28 c6 ->
    ../../devices/pci@780/pci@0/pci@8/SUNW,qlc@0,1/fp@0,0:fc
```

注 - 控制器 c5 和 c6 是双端口 FC HBA 上的端口 A 和 B。控制器 c1 和 c3 是单端口 SAS HBA 的端口。控制器 c2 是 Oracle 的 Sun Fire™ T2000 服务器中的内部 SAS 控制器。

确定希望显式启用或禁用多路径的端口。

3. 将 `/kernel/drv/fp.conf` 文件复制到 `/etc/driver/drv/fp.conf` 文件。
4. 选择以下方法之一来启用或禁用特定 FC HBA 端口：
 - 对于要在 `/etc/driver/drv/fp.conf` 中启用的每个 FC HBA 端口，请添加以下行：


```
name="fp" parent="parent-name" port=port-number mpzio-disable="no";
```

其中，*parent-name* 是端口设备名称，*port-number* 是 FC HBA 端口号。

例如，以下条目将在除两个指定端口外的所有 FC HBA 控制器端口禁用多路径：

```
mpxio-disable="yes";
name="fp" parent="/pci@6,2000/SUNW,qlc@2" port=0 mpxio-disable="no";
name="fp" parent="/pci@13,2000/pci@2/SUNW,qlc@5" port=0 mpxio-disable="no";
```

- 对于要禁用的每个 FC HBA 端口，请添加以下行：

```
name="fp" parent="parent-name" port=port-number mpxio-disable="yes";
```

例如：

```
name="fp" parent="/pci@6,2000/SUNW,qlc@2" port=0 mpxio-disable="yes";
```

5. 开始重新引导和配置过程。

```
# stmsboot -u
```

系统将提示您重新引导。在重新引导期间，`/etc/vfstab` 文件和转储设备配置将更新，以反映设备名称更改。

6. (可选) 重新引导之后，如有必要，按“[多路径注意事项](#)”[17]中所述，将您的应用程序配置为使用新的设备名称。

配置第三方存储设备

注 - 在配置任何第三方设备之前，确保该设备受到支持。有关正确的供应商和产品 ID、模式以及设备使用多路径软件所必需的各种设置的信息，请参阅第三方用户文档或咨询第三方供应商。

第三方设备配置注意事项

在为第三方设备配置多路径之前，需要了解以下内容：

- 设备必须支持 `REPORT_LUNS` SCSI 命令和 `SCSI-3 INQUIRY` 命令 VPD 设备识别页 (0x83)。
- 您需要设备的供应商 ID (vendor ID, VID) 和产品 ID (product ID, PID)。通过在您的系统上使用后面跟有查询选项的 `format` 命令，可获得这些内容。有关更多信息，请参见 [format\(1M\)](#)。

启用多路径之后，多路径访问仍依赖于接受设备的特定于设备的 `scsi_vhci` 故障转移实现。缺省情况下，在每个故障转移实现中为 `scsi_vhci` 代码自动调用探测函数，从而查找表明设备受支持的第一个探测结果。

探测实现根据 `scsi_inquiry(9S)` 数据的某些组合来确定支持。包含表明符合 T10 TPGS (Target-Port-Group-Support, 目标端口组支持) 标准 INQUIRY 数据的设备将使用基于标准的 TPGS 故障转移实现。对于非符合设备, 故障转移实现的探测通常根据 VID/PID 匹配 (对照专用编译表) 确定支持。

为覆盖探测进程, `scsi_vhci.conf` 文件支持 `scsi-vhci-failover-override` 属性。`scsi-vhci-failover-override` 的值可用于为探测当前不接受的设备建立支持、覆盖探测支持或为设备禁用多路径支持。

配置第三方存储设备：新设备

可在第三方对称存储设备上配置多路径。对称存储设备是这样的存储设备, 指向该存储设备的所有路径均已激活, 且通过任一路径都可发出 I/O 命令。

▼ 如何配置第三方设备

如果您的系统已启用多路径, 请执行以下步骤配置第三方设备。如果您的系统已禁用多路径, 可在启用多路径 (如本章前文所述) 的同时配置第三方设备。

1. 成为管理员。
2. 将 `/kernel/drv/scsi_vhci.conf` 文件复制到 `/etc/driver/drv/scsi_vhci.conf` 文件。
3. 编辑 `/etc/driver/drv/scsi_vhci.conf` 文件, 添加供应商 ID 和产品 ID 条目。

供应商 ID 和产品 ID 是设备在 SCSI INQUIRY 数据中返回的供应商和产品标识字符串。供应商 ID 的长度必须为 8 个字符。您必须指定全部 8 个字符, 即使结尾字符是空格。

产品 ID 的长度最多可为 16 个字符。

```
scsi-vhci-failover-override =  
"VendorID1ProductID1", "f_sym",  
"VendorID2ProductID2", "f_sym",  
...  
"VendorIDnProductIDn", "f_sym";
```

请注意, 各个条目由 ';' 字符 (逗号) 分隔, 最后一个供应商/产品条目以 ';' 字符 (分号) 结尾。

例如, 要添加供应商为 "ACME"、产品 ID 为 "MSU" 的设备以及供应商为 "XYZ"、产品 ID 为 "ABC" 的设备, 应向 `/etc/driver/drv/scsi_vhci.conf` 文件中添加以下行:

```
scsi-vhci-failover-override =  
"ACME    MSU", "f_sym",  
"XYZ     ABC", "f_sym";
```

4. 保存 `/etc/driver/drv/scsi_vhci.conf` 文件, 然后退出。

5. 开始重新引导和配置过程。

```
# stmsboot -u
```

系统将提示您重新引导。在重新引导期间，`/etc/vfstab` 文件和转储配置将更新，以反映设备名称更改。

6. 如有必要，按照[“启用和禁用多路径” \[18\]](#)中所述执行设备名称更新。

配置第三方存储设备：禁用设备

可为使用特定供应商 ID/产品 ID 组合的所有设备禁用多路径。需要在 `scsi_vhci.conf` 文件中指定这一排除。

▼ 如何禁用第三方设备

1. 成为管理员。
2. 将 `/kernel/drv/scsi_vhci.conf` 文件复制到 `/etc/driver/drv/scsi_vhci.conf` 文件。
3. 向 `/etc/driver/drv/scsi_vhci.conf` 文件中添加供应商 ID 和产品 ID 条目。

供应商 ID 和产品 ID 是设备在 SCSI INQUIRY 数据中返回的供应商和产品标识字符串。供应商 ID 的长度必须为 8 个字符。您必须指定全部 8 个字符，即使结尾字符是空格。产品 ID 的长度最多可为 16 个字符。

```
scsi-vhci-failover-override =
"VendorID1ProductID1", "NONE",
"VendorID2ProductID2", "NONE",
...
"VendorIDnProductIDn", "NONE";
```

以上示例中的各个条目由 ';' 字符（逗号）分隔，最后一个供应商/产品条目以 ';' 字符（分号）结尾。例如，要添加供应商为 "ACME"、产品 ID 为 "MSU" 的设备以及供应商为 "XYZ"、产品 ID 为 "ABC" 的供应商设备，应向 `/etc/driver/drv/scsi_vhci.conf` 文件中添加以下行：

```
scsi-vhci-failover-override =
"ACME    MSU", "NONE",
"XYZ     ABC", "NONE";
```

4. 保存 `scsi_vhci.conf` 文件，然后退出。
5. 开始重新引导和配置过程。

```
# stmsboot -u
```

系统将提示您重新引导。在重新引导期间，`/etc/vfstab` 文件和转储配置将更新，以反映设备名称更改。

6. 如有必要，按照[“启用和禁用多路径” \[18\]](#)中所述执行设备名称更新。

显示设备名称更改

对多路径配置进行更改后，可通过调用带 `-e`、`-d` 或 `-u` 选项的 `stmsboot` 命令来显示非多路径与多路径设备名称之间的映射。非多路径和多路径设备名称必须都存在，这样才能显示映射。

显示所有控制器上设备的映射。例如：

```
# stmsboot -L
non-STMS device name          STMS device name
-----
/dev/rdisk/c2t8d0             /dev/rdisk/c10t500000E01046DEE0d0
/dev/rdisk/c2t0d0             /dev/rdisk/c10t500000E01046B070d0
/dev/rdisk/c2t3d0             /dev/rdisk/c10t20000020372A40AFd0
/dev/rdisk/c2t12d0            /dev/rdisk/c10t500000E01046DEF0d0
/dev/rdisk/c2t11d0            /dev/rdisk/c10t500000E01046E390d0
/dev/rdisk/c3t8d0             /dev/rdisk/c10t500000E01046DEE0d0
/dev/rdisk/c3t0d0             /dev/rdisk/c10t500000E01046B070d0
/dev/rdisk/c3t3d0             /dev/rdisk/c10t20000020372A40AFd0
/dev/rdisk/c3t12d0            /dev/rdisk/c10t500000E01046DEF0d0
/dev/rdisk/c3t11d0            /dev/rdisk/c10t500000E01046E390d0
```

`-l` 选项仅显示指定控制器上设备的映射。以下示例显示控制器 3 的映射。

```
# stmsboot -l3
non-STMS device name          STMS device name
-----
/dev/rdisk/c3t8d0             /dev/rdisk/c10t500000E01046DEE0d0
/dev/rdisk/c3t0d0             /dev/rdisk/c10t500000E01046B070d0
/dev/rdisk/c3t3d0             /dev/rdisk/c10t20000020372A40AFd0
/dev/rdisk/c3t12d0            /dev/rdisk/c10t500000E01046DEF0d0
/dev/rdisk/c3t11d0            /dev/rdisk/c10t500000E01046E390d0
```

配置自动故障恢复

某些存储设备在配置阵列期间已将控制器配置为“主”和“辅助”。辅助路径在运行时，其性能水平要低于主路径。多路径软件使用主路径与存储设备通信，并使辅助路径处于待机状态。

如果主路径出现故障，多路径软件会自动将所有 I/O 通信引向辅助路径，同时主路径处于脱机状态。此进程称为“故障转移”操作。如果与主路径关联的故障已经修复，多路径

软件会像先前那样，自动将所有 I/O 通信引向主路径，同时使辅助路径处于待机状态。此进程称为故障恢复操作。

可禁用自动故障恢复操作，这样多路径软件不会自动故障恢复到主路径。以后，在与主路径关联的故障修复后，可使用 `luxadm` 命令执行手动故障恢复操作。有关更多信息，请参见 [luxadm\(1M\)](#)。

▼ 如何配置自动故障恢复

1. 成为管理员。
2. 将 `/kernel/drv/scsi_vhci.conf` 文件复制到 `/etc/driver/drv/scsi_vhci.conf` 文件。
3. 在 `/kernel/drv/scsi_vhci.conf` 文件中，通过更改 `auto-failback` 条目来启用或禁用自动故障恢复功能。

```
auto-failback="enable";

auto-failback="disable";
```

4. 保存并退出该文件。
5. 重新引导系统。

```
# shutdown -g0 -y -i6
```

管理 Solaris I/O 多路径

使用 `mpathadm` 命令确定和配置 Solaris OS 多路径支持。`mpathadm` 命令通过 ANSI 标准多路径管理 API 启用多路径管理。本章中用来表示路径、启动器端口、目标端口和 LU 的术语与 T10 规范一致。

执行以下任务来管理多路径设备。

- “[显示多路径支持信息](#)” [27]
- “[为多路径支持配置自动故障恢复](#)” [35]

显示多路径支持信息

可以使用 `mpathadm` 命令显示多路径支持信息并管理多路径搜索。多路径支持和属性信息用多路径管理 API 插件库名称进行标识，可使用 `mpathadm` 命令显示该插件库名称。

▼ 如何显示多路径支持信息

1. 成为管理员。
2. 确定系统上的多路径支持。

```
# mpathadm list mpath-support
mpath-support: libmpscsi_vhci.so
```

3. 显示指定 **mpath** 支持名称的多路径支持属性。

```
# mpathadm show mpath-support libmpscsi_vhci.so
mpath-support: libmpscsi_vhci.so
Vendor: Sun Microsystems
Driver Name: scsi_vhci
Default Load Balance: round-robin
Supported Load Balance Types:
    round-robin
    logical-block
Allows To Activate Target Port Group Access: yes
Allows Path Override: no
Supported Auto Failback Config: yes
Auto Failback: on
Failback Polling Rate (current/max): 0/0
Supported Auto Probing Config: 0
Auto Probing: NA
Probing Polling Rate (current/max): NA/NA
Supported Devices:
Vendor: SUN
Product: T300
Revision:
Supported Load Balance Types:
    round-robin

Vendor: SUN
Product: T4
Revision:
Supported Load Balance Types:
    round-robin

.
.
.
```

命令输出还会显示多路径支持软件支持的设备产品的列表。缺省情况下，多路径支持 libmpscsi_vhci.so 库文件支持符合 T10 目标端口组的设备。

▼ 如何显示特定启动器端口的属性

以下步骤描述如何显示启动器端口的属性。

1. 列出启动器端口。

```
# mpathadm list initiator-port
Initiator Port:  iqn.1986-03.com.sun:01:ffffffffffff.4e94f9bd,4000002a00ff
Initiator Port:  210100e08ba41feb
Initiator Port:  210000e08b841feb
```

2. 显示特定启动器端口的属性。

```
# mpathadm show initiator-port 2000000173018713
Initiator Port:  210100e08ba41feb
      Transport Type:  Fibre Channel
      OS Device File:  /devices/pci@1,0/pci1022,7450@1/pci1077,141@2,1/fp@0,0
```

如果不使用 `mpathadm show initiator-port` 命令指定启动器端口，则将显示所有发现的启动器端口的属性。

▼ 如何显示特定 LU 信息

使用 `mpathadm` 命令显示 LU 列表以及每个 LU 的属性。显示的 LU 列表中又包含可用于显示特定 LU 的属性的名称。

1. 显示多路径 LU 的列表。

```
# mpathadm list lu
/dev/rdisk/c0t600144F08069703400004E828EE10004d0s2
      Total Path Count: 8
      Operational Path Count: 8
/dev/rdisk/c0t600144F08069703400004E8183DF0002d0s2
      Total Path Count: 8
      Operational Path Count: 8
/dev/rdisk/c0t600A0B800026D63A0000A4994E2342D4d0s2
      Total Path Count: 4
      Operational Path Count: 4
/dev/rdisk/c0t600A0B800029065C00007CF54E234013d0s2
      Total Path Count: 4
      Operational Path Count: 4
/dev/rdisk/c0t600A0B800026D63A0000A4984E234298d0s2
      Total Path Count: 4
      Operational Path Count: 4
/dev/rdisk/c0t600A0B800029065C00007CF44E233FCFd0s2
      Total Path Count: 4
      Operational Path Count: 4
/dev/rdisk/c0t600A0B800026D63A0000A4974E23424Ed0s2
      Total Path Count: 4
      Operational Path Count: 4
.
.
.
```

2. 显示使用列表中 LU 名称的特定 LU 的配置信息。

```
# mpathadm show lu /dev/rdisk/c4t60003BA27D2120004204AC2B000DAB00d0s2
```

```
mpath-support: libmpscsi_vhci.so
Vendor: SUN
Product: Sun Storage NAS
Revision: 1.0
Name Type: unknown type
Name: 600144f08069703400004e828ee10004
Asymmetric: yes
Current Load Balance: round-robin
Logical Unit Group ID: NA
Auto Failback: on
Auto Probing: NA
```

Paths:

```
Initiator Port Name: 210100e08ba41feb
Target Port Name: 2100001b329b6c3f
Override Path: NA
Path State: OK
Disabled: no
```

```
Initiator Port Name: 210100e08ba41feb
Target Port Name: 2101001b32bb6c3f
Override Path: NA
Path State: OK
Disabled: no
```

```
Initiator Port Name: 210100e08ba41feb
Target Port Name: 2100001b329b793c
Override Path: NA
Path State: OK
Disabled: no
```

.
.
.

Target Port Groups:

```
ID: 0
Explicit Failover: no
Access State: active optimized
Target Ports:
    Name: 2100001b329b6c3f
    Relative ID: 1
```

```
    Name: 2101001b32bb6c3f
    Relative ID: 2
```

```
ID: 1
Explicit Failover: no
Access State: standby
Target Ports:
    Name: 2100001b329b793c
    Relative ID: 257
```

```
    Name: 2101001b32bb793c
    Relative ID: 256
```

如果不使用 `mpathadm show lu` 命令指定 LU 名称，则将显示有关所有发现的逻辑单元的信息。

▼ 如何显示与特定目标端口相关联的所有 LU

使用以下步骤显示通过目标端口的路径。

1. 显示 LU 列表。

```
# mpathadm list lu
/dev/rdisk/c0t600144F08069703400004E828EE10004d0s2
    Total Path Count: 8
    Operational Path Count: 8
/dev/rdisk/c0t600144F08069703400004E8183DF0002d0s2
    Total Path Count: 8
    Operational Path Count: 8
/dev/rdisk/c0t600A0B800026D63A0000A4994E2342D4d0s2
    Total Path Count: 4
    Operational Path Count: 4
/dev/rdisk/c0t600A0B800029065C00007CF54E234013d0s2
    Total Path Count: 4
    Operational Path Count: 4
.
.
.
```

2. 显示特定 LU 的信息，以确定目标端口。

```
# mpathadm show lu /dev/rdisk/c0t600A0B800029065C00007CF54E234013d0s2
Logical Unit: /dev/rdisk/c0t600A0B800029065C00007CF54E234013d0s2
    mpath-support: libmpscsi_vhci.so
    Vendor: SUN
    Product: CSM200_R
    Revision: 0660
    Name Type: unknown type
    Name: 600a0b800029065c00007cf54e234013
    Asymmetric: yes
    Current Load Balance: round-robin
    Logical Unit Group ID: NA
    Auto Failback: on
    Auto Probing: NA

Paths:
    Initiator Port Name: 210000e08b841feb
    Target Port Name: 200800a0b826d63b
    Override Path: NA
    Path State: OK
    Disabled: no

    Initiator Port Name: 210000e08b841feb
    Target Port Name: 200900a0b826d63b
```

```
Override Path: NA
Path State: OK
Disabled: no
.
.
.

Target Port Groups:
  ID: 4
  Explicit Failover: yes
  Access State: standby
  Target Ports:
    Name: 200800a0b826d63b
    Relative ID: 0

    Name: 200800a0b826d63c
    Relative ID: 0

  ID: 14
  Explicit Failover: yes
  Access State: active
  Target Ports:
    Name: 200900a0b826d63b
    Relative ID: 0

    Name: 200900a0b826d63c
    Relative ID: 0
```

3. 显示特定的目标端口信息。

```
# mpathadm list lu -t 20030003ba27d212
mpath-support: libmpscsi_vhci.so
/dev/rdisk/c0t600A0B800026D63A0000A4994E2342D4d0s2
  Total Path Count: 4
  Operational Path Count: 4
mpath-support: libmpscsi_vhci.so
/dev/rdisk/c0t600A0B800029065C00007CF54E234013d0s2
  Total Path Count: 4
  Operational Path Count: 4
mpath-support: libmpscsi_vhci.so
/dev/rdisk/c0t600A0B800026D63A0000A4984E234298d0s2
  Total Path Count: 4
  Operational Path Count: 4
mpath-support: libmpscsi_vhci.so
/dev/rdisk/c0t600A0B800029065C00007CF44E233FCFd0s2
  Total Path Count: 4
  Operational Path Count: 4
.
.
.
```

▼ 如何显示特定名称的 LU

显示有关多路径 LU 的详细信息，包括路径和目标端口组信息。请注意，信息中的名称属性表示该 LU 的标识符，它是从硬件派生出来的，并由该系统使用。如果此名称源自 SCSI Inquiry 重要产品数据 (Vital Product Data, VPD) 页面 83h，则名称类型属性表示 SCSI 标准定义的一个关联标识符类型。

1. 显示多路径 LU 的列表。

```
# mpathadm list lu
/dev/rdisk/c0t600144F08069703400004E828EE10004d0s2
    Total Path Count: 8
    Operational Path Count: 8
/dev/rdisk/c0t600144F08069703400004E8183DF0002d0s2
    Total Path Count: 8
    Operational Path Count: 8
/dev/rdisk/c0t600A0B800026D63A0000A4994E2342D4d0s2
    Total Path Count: 4
    Operational Path Count: 4
/dev/rdisk/c0t600A0B800029065C00007CF54E234013d0s2
    Total Path Count: 4
    Operational Path Count: 4
/dev/rdisk/c0t600A0B800026D63A0000A4984E234298d0s2
    Total Path Count: 4
    Operational Path Count: 4
/dev/rdisk/c0t600A0B800029065C00007CF44E233FCFd0s2
    Total Path Count: 4
    Operational Path Count: 4
.
.
.
```

2. 显示选定 LU 的配置信息。

```
# mpathadm show lu /dev/rdisk/c0t600A0B800026D63A0000A4994E2342D4d0s2
Logical Unit: /dev/rdisk/c0t600A0B800026D63A0000A4994E2342D4d0s2
mpath-support: libmpscsi_vhci.so
Vendor: SUN
Product: CSM200_R
Revision: 0660
Name Type: unknown type
Name: 600a0b800026d63a0000a4994e2342d4
Asymmetric: yes
Current Load Balance: round-robin
Logical Unit Group ID: NA
Auto Failback: on
Auto Probing: NA

Paths:
    Initiator Port Name: 210000e08b841feb
    Target Port Name: 200800a0b826d63b
    Override Path: NA
```

```
Path State: OK
Disabled: no

Initiator Port Name: 210000e08b841feb
Target Port Name: 200900a0b826d63b
Override Path: NA
Path State: OK
Disabled: no

Initiator Port Name: 210000e08b841feb
Target Port Name: 200800a0b826d63c
Override Path: NA
Path State: OK
Disabled: no

Initiator Port Name: 210000e08b841feb
Target Port Name: 200900a0b826d63c
Override Path: NA
Path State: OK
Disabled: no

Target Port Groups:
ID: 3
Explicit Failover: yes
Access State: active
Target Ports:
    Name: 200800a0b826d63b
    Relative ID: 0

    Name: 200800a0b826d63c
    Relative ID: 0

ID: 13
Explicit Failover: yes
Access State: standby
Target Ports:
    Name: 200900a0b826d63b
    Relative ID: 0

    Name: 200900a0b826d63c
    Relative ID: 0
```

可以使用 `mpathadm show lu` 命令指定 `-e` 或 `--path-missing` 选项以仅显示缺少的路径的逻辑单元。

3. 显示选定 LU 的信息。

```
# mpathadm list lu -n 600a0b800026d63a0000a4994e2342d4
mpath-support: libmpscsi_vhci.so
/dev/rdisk/c0t600A0B800026D63A0000A4994E2342D4d0s2
Total Path Count: 4
Operational Path Count: 4
```

为多路径支持配置自动故障恢复

对称设备可以自动故障恢复到可能的最优路径。如果在初始路径上发生了故障转移，则备用路径将成为新的联机路径。通常，备用路径不是最理想的路径。当启用自动故障恢复时，初始路径会重新联机，并自动故障转移到初始路径。

▼ 如何为特定多路径支持配置自动故障恢复

1. 成为管理员。
2. 显示受支持的多路径驱动程序信息。

```
# mpathadm list mpath-support
mpath-support: libmpscsi_vhci.so
```

3. 为受支持的多路径驱动程序启用自动故障恢复支持。

```
# mpathadm modify mpath-support -a on libmpscsi_vhci.so
```

4. 确认配置更改。

```
# mpathadm show mpath-support libmpscsi_vhci.so
mpath-support: libmpscsi_vhci.so
    Vendor: Sun Microsystems
    Driver Name: scsi_vhci
    Default Load Balance: round-robin
    Supported Load Balance Types:
        round-robin
        logical-block
    Allows To Activate Target Port Group Access: yes
    Allows Path Override: no
    Supported Auto Failback Config: 1
    Auto Failback: on
    Failback Polling Rate (current/max): 0/0
    Supported Auto Probing Config: 0
    Auto Probing: NA
    Probing Polling Rate (current/max): NA/NA
    Supported Devices:
        Vendor: SUN
        Product: T300
        Revision:
        Supported Load Balance Types:
            round-robin

        Vendor: SUN
        Product: T4
        Revision:
        Supported Load Balance Types:
            round-robin
```

.

注 - 当系统正在运行时，`mpathadm modify` 命令设置所启动的自动显示有效。但是，要保持更改的设置持久，必须更新 `/etc/driver/drv/scsi_vhci.conf` 文件。请参阅[“配置自动故障恢复” \[26\]](#)。

▼ 如何对 LU 进行故障转移

此操作仅适用于以下两个类别的设备：

- 具有由多路径支持可识别和支持的专用故障转移机制的非对称设备
- 符合 T10 标准目标端口组支持 `libmpscsi_vhci.so` 并提供了显式模式非对称 LU 访问的设备

1. 显示多路径 LU 的列表。

```
# mpathadm list lu
/dev/rdisk/c0t600144F08069703400004E828EE10004d0s2
    Total Path Count: 8
    Operational Path Count: 8
/dev/rdisk/c0t600144F08069703400004E8183DF0002d0s2
    Total Path Count: 8
    Operational Path Count: 8
/dev/rdisk/c0t600A0B800026D63A0000A4994E2342D4d0s2
    Total Path Count: 4
    Operational Path Count: 4
/dev/rdisk/c0t600A0B800029065C00007CF54E234013d0s2
    Total Path Count: 4
    Operational Path Count: 4
/dev/rdisk/c0t600A0B800026D63A0000A4984E234298d0s2
    Total Path Count: 4
    Operational Path Count: 4
.
.
.
```

2. 显示特定 LU 的配置信息。

```
# mpathadm show lu /dev/rdisk/c0t600A0B800026D63A0000A4984E234298d0s2
Logical Unit: /dev/rdisk/c0t600A0B800026D63A0000A4984E234298d0s2
mpath-support: libmpscsi_vhci.so
Vendor: SUN
Product: CSM200_R
Revision: 0660
Name Type: unknown type
Name: 600a0b800026d63a0000a4984e234298
Asymmetric: yes
Current Load Balance: round-robin
Logical Unit Group ID: NA
```

```

Auto Failback: on
Auto Probing: NA

Paths:
    Initiator Port Name: 210000e08b841feb
    Target Port Name: 200800a0b826d63b
    Override Path: NA
    Path State: OK
    Disabled: no

    Initiator Port Name: 210000e08b841feb
    Target Port Name: 200900a0b826d63b
    Override Path: NA
    Path State: OK
    Disabled: no

    Initiator Port Name: 210000e08b841feb
    Target Port Name: 200800a0b826d63c
    Override Path: NA
    Path State: OK
    Disabled: no

    Initiator Port Name: 210000e08b841feb
    Target Port Name: 200900a0b826d63c
    Override Path: NA
    Path State: OK
    Disabled: no

```

```

Target Port Groups:
    ID: 5
    Explicit Failover: yes
    Access State: active
    Target Ports:
        Name: 200800a0b826d63b
        Relative ID: 0

        Name: 200800a0b826d63c
        Relative ID: 0

    ID: 15
    Explicit Failover: yes
    Access State: standby
    Target Ports:
        Name: 200900a0b826d63b
        Relative ID: 0

        Name: 200900a0b826d63c
        Relative ID: 0

```

3. 手动强制执行 LU 故障转移。

```
# mpathadm failover lu /dev/rdisk/c0t600A0B800026D63A0000A4984E234298d0s2
```

如果此操作成功，则由于逻辑单元故障转移，设备的目标端口组的访问状态将改变。

4. 确认访问状态更改。

```
# mpathadm show lu /dev/rdisk/c0t600A0B800026D63A0000A4984E234298d0s2
Logical Unit: /dev/rdisk/c0t600A0B800026D63A0000A4984E234298d0s2
  mpath-support: libmpscsi_vhci.so
  Vendor: SUN
  Product: CSM200_R
  Revision: 0660
  Name Type: unknown type
  Name: 600a0b800026d63a0000a4984e234298
  Asymmetric: yes
  Current Load Balance: round-robin
  Logical Unit Group ID: NA
  Auto Failback: on
  Auto Probing: NA

Paths:

  Initiator Port Name: 210000e08b841feb
  Target Port Name: 200800a0b826d63b
  Override Path: NA
  Path State: OK
  Disabled: no

  Initiator Port Name: 210000e08b841feb
  Target Port Name: 200900a0b826d63b
  Override Path: NA
  Path State: OK
  Disabled: no

  Initiator Port Name: 210000e08b841feb
  Target Port Name: 200800a0b826d63c
  Override Path: NA
  Path State: OK
  Disabled: no

  Initiator Port Name: 210000e08b841feb
  Target Port Name: 200900a0b826d63c
  Override Path: NA
  Path State: OK
  Disabled: no

Target Port Groups:
  ID: 5
  Explicit Failover: yes
  Access State: standby
  Target Ports:
    Name: 200800a0b826d63b
    Relative ID: 0

    Name: 200800a0b826d63c
    Relative ID: 0

  ID: 15
  Explicit Failover: yes
```

```

Access State: active
Target Ports:
    Name: 200900a0b826d63b
    Relative ID: 0

    Name: 200900a0b826d63c
    Relative ID: 0

```

▼ 如何启用 LU 路径

如果指向 LU 的路径被禁用，则 `enable` 命令可将路径重新更改为启用。必须使用启动器端口名称、目标端口名称和 LU 指定完整路径。要验证更改，请对逻辑单元运行 `show` 命令。

1. 显示多路径 LU 的列表。

```

# mpathadm list lu
/dev/rdisk/c0t600144F08069703400004E828EE10004d0s2
    Total Path Count: 8
    Operational Path Count: 8
/dev/rdisk/c0t600144F08069703400004E8183DF0002d0s2
    Total Path Count: 8
    Operational Path Count: 8
/dev/rdisk/c0t600A0B800026D63A0000A4994E2342D4d0s2
    Total Path Count: 4
    Operational Path Count: 4
/dev/rdisk/c0t600A0B800029065C00007CF54E234013d0s2
    Total Path Count: 4
    Operational Path Count: 4
/dev/rdisk/c0t600A0B800026D63A0000A4984E234298d0s2
    Total Path Count: 4
    Operational Path Count: 4
/dev/rdisk/c0t600A0B800029065C00007CF44E233FCFd0s2
    Total Path Count: 4
    Operational Path Count: 4
.
.
.

```

2. 显示选定 LU 的配置信息。

```

# mpathadm show lu
Logical Unit: /dev/rdisk/c0t600A0B800026D63A0000A4984E234298d0s2
mpath-support: libmpscsi_vhci.so
Vendor: SUN
Product: CSM200_R
Revision: 0660
Name Type: unknown type
Name: 600a0b800026d63a0000a4984e234298
Asymmetric: yes
Current Load Balance: round-robin

```

```
Logical Unit Group ID:  NA
Auto Failback:  on
Auto Probing:  NA

Paths:

    Initiator Port Name:  210000e08b841feb
    Target Port Name:  200800a0b826d63b
    Override Path:  NA
    Path State:  OK
    Disabled:  no

    Initiator Port Name:  210000e08b841feb
    Target Port Name:  200900a0b826d63b
    Override Path:  NA
    Path State:  OK
    Disabled:  no

    Initiator Port Name:  210000e08b841feb
    Target Port Name:  200800a0b826d63c
    Override Path:  NA
    Path State:  OK
    Disabled:  no

    Initiator Port Name:  210000e08b841feb
    Target Port Name:  200900a0b826d63c
    Override Path:  NA
    Path State:  OK
    Disabled:  no

Target Port Groups:
    ID:  5
    Explicit Failover:  yes
    Access State:  standby
    Target Ports:
        Name:  200800a0b826d63b
        Relative ID:  0

        Name:  200800a0b826d63c
        Relative ID:  0

    ID:  15
    Explicit Failover:  yes
    Access State:  active
    Target Ports:
        Name:  200900a0b826d63b
        Relative ID:  0

        Name:  200900a0b826d63c
        Relative ID:  0
```

3. 启用 LU 路径。

```
# mpathadm enable path -i 210000e08b841feb -t 200900a0b826d63b \
-l /dev/rdisk/c0t600A0B800026D63A0000A4984E234298d0s2
```

▼ 如何禁用 LU 路径

此操作禁用 LU 路径，而不管其操作状态是什么。

注 - 禁用状态不会在重新引导之间持续存在。如果在下一个引导序列之前路径是可操作的，则缺省情况下将启用该路径。当给定路径是剩余的最后一个可操作路径时，不允许执行此操作。

1. 显示多路径 LU 的列表。

```
# mpathadm list lu
/dev/rdisk/c0t600144F08069703400004E828EE10004d0s2
    Total Path Count: 8
    Operational Path Count: 8
/dev/rdisk/c0t600144F08069703400004E8183DF0002d0s2
    Total Path Count: 8
    Operational Path Count: 8
/dev/rdisk/c0t600A0B800026D63A0000A4994E2342D4d0s2
    Total Path Count: 4
    Operational Path Count: 4
/dev/rdisk/c0t600A0B800029065C00007CF54E234013d0s2
    Total Path Count: 4
    Operational Path Count: 4
/dev/rdisk/c0t600A0B800026D63A0000A4984E234298d0s2
    Total Path Count: 4
    Operational Path Count: 4
/dev/rdisk/c0t600A0B800029065C00007CF44E233FCFd0s2
    Total Path Count: 4
    Operational Path Count: 4
.
.
.
```

2. 显示特定 LU 的配置信息。

```
# mpathadm show lu /dev/rdisk/c0t600A0B800029065C00007CF34E233F89d0s2
Logical Unit: /dev/rdisk/c0t600A0B800029065C00007CF34E233F89d0s2
mpath-support: libmpscsi_vhci.so
Vendor: SUN
Product: CSM200_R
Revision: 0660
Name Type: unknown type
Name: 600a0b800029065c00007cf34e233f89
Asymmetric: yes
Current Load Balance: round-robin
Logical Unit Group ID: NA
Auto Failback: on
Auto Probing: NA

Paths:
    Initiator Port Name: 210000e08b841feb
```

```
Target Port Name: 200800a0b826d63b
Override Path: NA
Path State: OK
Disabled: no

Initiator Port Name: 210000e08b841feb
Target Port Name: 200900a0b826d63b
Override Path: NA
Path State: OK
Disabled: no

Initiator Port Name: 210000e08b841feb
Target Port Name: 200800a0b826d63c
Override Path: NA
Path State: OK
Disabled: no

Initiator Port Name: 210000e08b841feb
Target Port Name: 200900a0b826d63c
Override Path: NA
Path State: OK
Disabled: no
```

```
Target Port Groups:
ID: 8
Explicit Failover: yes
Access State: standby
Target Ports:
    Name: 200800a0b826d63b
    Relative ID: 0

    Name: 200800a0b826d63c
    Relative ID: 0

ID: 18
Explicit Failover: yes
Access State: active
Target Ports:
    Name: 200900a0b826d63b
    Relative ID: 0

    Name: 200900a0b826d63c
    Relative ID: 0
```

3. 选择启动器端口和目标端口名称。
4. 禁用选定的 LU 路径。

```
# mpathadm disable path -i 210000e08b841feb -t 200900a0b826d63b \
-l /dev/rdsd/c0t600A0B800029065C00007CF34E233F89d0s2
```

配置光纤连接设备

本章简要介绍了用于配置光纤连接 (fabric-connected, FC) 设备的步骤。在 Oracle Solaris OS 中，FCAL、光纤和点对点连接的设备可自动供系统使用。如果要手动配置光纤连接设备，请参见[附录 A, 光纤连接的设备的手动配置](#)。

本章包含以下主题：

- “FC 设备注意事项” [43]
- “添加 FC 设备” [43]
- “在 SPARC 上配置光纤引导设备” [44]

FC 设备注意事项

在配置 Solaris I/O 多路径功能之前，请查看以下注意事项：

- 按照供应商特定文档为存储和交换机配置端口和区域。
- 不再需要手动配置设备。
- LUN 屏蔽允许将特定 LUN 显示给特定系统。请参见介绍屏蔽的供应商特定存储文档。
- 将阵列和其他存储设备连接到具有或不具有多路径功能的 SAN。Solaris 多路径是一个关联的应用程序，它与产品名捆绑在一起。
- STMS 引导实用程序包括在管理 SAN 引导进程的 Solaris I/O 多路径功能中。发出 `stmsboot` 命令将激活 `/etc/vfstab` 和转储配置的自动更新，以在启用或禁用多路径软件时反映设备名称更改。请注意，缺省情况下，该软件对于基于 SPARC 的系统上的设备处于禁用状态，对于基于 x86 的系统上的设备处于启用状态。

添加 FC 设备

下表列出了添加和删除 FC 设备时的常用命令。

命令	说明	详细信息
cfgadm 和 cfgadm_fp	动态重新配置设备和 FC 设备。这两个命令最常用于配置 SAN 上的存储设备。	cfgadm(1M) 和 cfgadm_fp(1M)
fcinfo	收集有关主机上光纤通道主机总线适配器 (Host Bus Adapter, HBA) 端口的管理信息。	fcinfo(1M)
format	标识与系统相连接的设备并能够为磁盘添加标签。	format(1M)
luxadm	管理存储设备和 FC_AL 设备。	luxadm(1M)

注 - 如果在启用多路径功能时使用 format 命令，您只能看到每个 LUN 的设备标识符的一个实例。如果没有启用多路径功能，对于每个路径您将看到一个标识符。

▼ 如何添加 FC 设备

1. 在 FC 设备上创建所需的 LUN。
2. 如有必要，在 FC 设备上为 HBA 控件应用 LUN 屏蔽。
3. 将存储设备连接到系统。
4. 如必要，为 FC 设备上的交换机创建基于端口的区域或 WWN 区域。
5. 在存储设备的 LUN 或磁盘组上挂载可用的任何现有文件系统。

在 SPARC 上配置光纤引导设备

可以将运行 Oracle Solaris OS 的 SPARC 服务器设置为从光纤磁盘设备引导。

光纤引导设备注意事项

光纤引导设备可通过 Oracle Solaris 安装方法来访问，如同在先前的 Solaris OS 发行版中访问内部引导设备那样。有关详细信息，请参阅 Oracle Solaris 安装指南。启用光纤引导设备时请注意以下事项：

- 通过以下措施将对引导设备的干扰降至最低：
 - 确保引导设备不是过度订阅的目标或 LUN

- 避免在目标或 LUN 上安装应用程序和软件
- 减少主机与光纤设备之间的物理距离及转发数
- 开始光纤引导过程之前，从卷管理器控件移除引导磁盘。
- 确保在系统上为 HBA 装入最新 HBA fcode 和驱动程序。
- 如果引导设备上需要多路径，请按[第 2 章 配置 Solaris I/O 多路径功能](#)中的介绍使用 `stmsboot` 命令。

配置 Solaris iSCSI 启动器

本章介绍如何配置 Oracle Solaris 发行版中的 Solaris iSCSI 启动器。有关与配置 iSCSI 启动器关联的过程的信息，请参见[“配置 iSCSI 启动器任务” \[48\]](#)。

本章包含以下主题：

- [“Oracle Solaris iSCSI 技术（概述）” \[47\]](#)
- [“建议的 iSCSI 配置做法” \[49\]](#)
- [“配置 iSCSI 启动器任务” \[48\]](#)
- [“在基于 iSCSI 的存储网络中配置验证” \[54\]](#)
- [“设置 Oracle Solaris iSCSI 多路径设备” \[59\]](#)
- [“监视 iSCSI 配置” \[62\]](#)
- [“修改 iSCSI 启动器和目标参数” \[64\]](#)
- [“iSCSI 配置问题的故障排除” \[70\]](#)

有关使用 COMSTAR 配置 Oracle Solaris iSCSI 目标的信息，请参见《[在 Oracle Solaris 11.2 中管理设备](#)》中的第 8 章“使用 COMSTAR 配置存储设备”。

有关安装和引导 iSCSI 磁盘的更多信息，请参见《[安装 Oracle Solaris 11.2 系统](#)》中的“[如何执行 GUI 安装](#)”中的步骤 7。

Oracle Solaris iSCSI 技术（概述）

iSCSI 是 Internet SCSI (Small Computer System Interface, 小型计算机系统接口) 的首字母缩略词，是用于链接数据存储子系统的基于 Internet 协议 (Internet Protocol, IP) 的存储网络标准。

通过在 IP 网络上传输 SCSI 命令，iSCSI 协议可用于通过网络访问块设备，就像这些设备连接至本地系统一样。

如果要使用现有 TCP/IP 网络中的存储设备，则可以使用以下解决方法：

- iSCSI 块设备或磁带 – 将 SCSI 命令和数据从块级别转换为 IP 数据包。如果需要在 一个系统与目标设备（如磁带设备或数据库）之间进行块级别访问，则在网络中使用 iSCSI 很有优势。对块级别设备的访问不会被锁定，因此您可以让多个用户或系统访问块级别设备（如 iSCSI 目标设备）。

- NFS – 通过 IP 传输文件数据。在网络中使用 NFS 的优点在于可以在多个系统之间共享文件数据。当很多用户访问 NFS 环境中可用的数据时，对文件数据的访问就会相应地被锁定。

以下是在 Oracle Solaris 上使用 iSCSI 目标和启动器的益处：

- iSCSI 协议可以跨现有以太网运行。
 - 可以使用任何支持的网络接口卡 (network interface card, NIC)、以太网集线器或以太网交换机。
 - 一个 IP 端口可以处理多个 iSCSI 目标设备。
 - 可以对 IP 网络使用现有的基础结构和管理工具。
- 可能具有可连接至客户机的现有光纤通道设备，而无需光纤通道 HBA 成本。此外，具有专用阵列的系统目前可以使用 Oracle Solaris ZFS 或 UFS 文件系统导出已复制的存储。
- 可以使用该协议连接至光纤通道或包含适当硬件的 iSCSI 存储区域网络 (Storage Area Network, SAN) 环境。

以下是在 Oracle Solaris 中使用 iSCSI 启动器软件时目前具有的限制或约束：

- 当前不支持使用 SLP 的 iSCSI 设备。
- 不能将 iSCSI 目标配置为转储设备。
- 在现有网络上传输大量数据可能对性能造成影响。
- iSCSI 启动器子系统仅在全局区域上运行，不在非全局区域上运行。从非全局区域内部访问 iSCSI 设备的解决方法是在全局区域中配置 iSCSI 启动器，并在区域配置过程中使用 `add device` 命令将设备添加到非全局区域。

确定 Oracle Solaris iSCSI 软件和硬件要求

- Oracle Solaris 存储软件和设备
 - 适用于提供存储设备的系统的 `group/feature/storage-server` 软件包
 - 用于 iSCSI 管理实用程序的 `system/storage/iscsi/iscsi-initiator` 软件包。
- 任何支持的 NIC

配置 iSCSI 启动器任务

以下是与配置 iSCSI 启动器关联的常规任务列表。某些任务是可选的，具体取决于您的网络配置需求。以下某些链接可转到介绍网络配置和目标配置的单独文档。

- [“确定 Oracle Solaris iSCSI 软件和硬件要求” \[48\]](#)
- [《在 Oracle Solaris 11.2 中管理 TCP/IP 网络、IPMP 和 IP 隧道》](#)
- [《在 Oracle Solaris 11.2 中管理设备》](#) 中的第 8 章 “使用 COMSTAR 配置存储设备”

- [“配置动态或静态目标搜索” \[50\]](#)
- [如何配置 iSCSI 启动器 \[51\]](#)
- [如何访问 iSCSI 磁盘 \[52\]](#)
- [“在基于 iSCSI 的存储网络中配置验证” \[54\]](#)
- [“设置 Oracle Solaris iSCSI 多路径设备” \[59\]](#)
- [“监视 iSCSI 配置” \[62\]](#)

iSCSI 术语

配置 iSCSI 目标和启动器之前，请查看以下术语。

术语	说明
搜索	搜索是指为启动器提供可用目标列表的过程。
搜索方法	指可找到 iSCSI 目标的方法。当前有三种可用方法： ■ Internet 存储名称服务 (Internet Storage Name Service, iSNS) – 通过与一个或多个 iSNS 服务器进行交互来搜索潜在目标。 ■ SendTargets – 通过使用 <i>discovery-address</i> 来搜索潜在目标。 ■ 静态 – 配置了静态目标寻址。
启动器	针对 iSCSI 目标启动 SCSI 请求的驱动程序。
启动器组	一组启动器。当启动器组与某个 LU 相关联时，只有该组的启动器可以访问该 LU。
iqn 或 eui 地址格式	iqn (iSCSI qualified name, iSCSI 限定名) 地址是 iSCSI 网络中的设备的唯一标识符，采用 <i>iqn.date.authority:uniqueid</i> 形式。当初初始化 iSCSI 启动器或目标时，将自动为 iSCSI 启动器或目标指定一个 IQN 名称。 eui (extended unique identifier, 扩展唯一标识符) 地址由 16 个十六进制数字组成，可用于标识一类既可在 SCSI 标准中又可在 Infini Band 标准中使用的 GUID。SRP 设备使用 eui 地址格式。
逻辑单元 (Logical Unit, LU)	存储系统中的唯一编号组件。当 LU 与一个或多个 SCSI 目标相关联时，该目标可以供一个或多个 SCSI 启动器访问。逻辑单元由逻辑单元号 (logical unit number, LUN) 标识。
目标设备	iSCSI 存储组件。
目标组	一组目标。一个 LU 可用于一个目标组中的所有目标。
目标门户组	确定特定 iSCSI 目标将侦听哪些接口的 IP 地址列表。TPG 包含 IP 地址和 TCP 端口号。

建议的 iSCSI 配置做法

在网络中配置 iSCSI 设备之前，请查看以下 iSCSI 建议。

- 设备

- 考虑使用多路径设备路径以增强可用性。
- 每个会话多个连接 (Multiple Connections per Session, MCS) 支持允许为同一个 iSCSI 会话使用多个从启动器到目标的 TCP/IP 连接。
- 网络
 - 在快速 (gigE 或更好) 专用网络上部署 iSCSI 设备。
 - 如果可能, 使用巨型帧 (jumbo frame) 以允许在每个以太网事务中传输更多数据, 从而减少帧数量。
 - 将 CAT6 标准电缆用于千兆网络基础结构。
 - 将 iSCSI 存储网络与局域网通信分离。
 - 配置多个会话或连接以在 TCP/IP 栈中使用多个线程。
 - 考虑 TCP 调优, 如禁用 Nagle 算法。
- 安全性
 - 除了物理安全性以外, 还使用 CHAP 验证, 该验证确保每个主机都有自己的口令。
 - 考虑使用 iSNS 目标搜索域, 该域可通过向使用自身的访问控制无法启用的目标提供访问控制, 同时将每个启动器的登录过程限制在网络中可用目标的相关子集范围内, 从而增强安全性。

配置动态或静态目标搜索

确定是要配置其中一种动态设备搜索方法, 还是要使用静态 iSCSI 启动器目标来执行设备搜索。

- 动态设备搜索 – 有两种可用的动态设备搜索方法：
 - SendTargets – 如果 iSCSI 节点显示了大量目标, 如 iSCSI 到光纤通道网桥, 则可以提供 iSCSI 节点 IP 地址/端口组合, 并允许 iSCSI 启动器使用 SendTargets 功能执行设备搜索。
 - iSNS – Internet 存储名称服务 (Internet Storage Name Service, iSNS) 允许 iSCSI 启动器搜索其有权访问的使用尽可能少的配置信息的目标。另外, 还提供状态更改通知功能, 在存储节点的操作状态发生更改时通知 iSCSI 启动器。要使用 iSNS 搜索选项, 可以提供 iSNS 服务器地址/端口组合, 并允许 iSCSI 启动器查询为执行设备搜索而指定的 iSNS 服务器。iSNS 服务器的缺省端口为 3205。有关 iSNS 的更多信息, 请参见 RFC 4171：
<http://www.ietf.org/rfc/rfc4171.txt>
iSNS 搜索服务提供一种管理模型, 用来搜索网络上的所有目标。
有关在 Oracle Solaris 中设置 iSNS 支持的更多信息, 请参见《在 Oracle Solaris 11.2 中管理设备》中的第 9 章“配置和管理 Oracle Solaris Internet 存储名称服务 (Internet Storage Name Service, iSNS)”。
- 静态设备搜索 – 如果 iSCSI 节点具有较少目标, 或者如果要限制启动器尝试访问的目标, 则可以使用以下静态目标地址命名约定以静态方式配置 *target-name*：

`target,target-address[:port-number]`

可以使用数组的管理工具来确定静态目标地址。

注 - 请勿将 iSCSI 目标配置为同时使用静态和动态设备搜索方法进行搜索。使用冗余搜索方法的后果是可能会在与 iSCSI 目标设备进行通信时减缓系统性能。

▼ 如何配置 iSCSI 启动器

启动器配置过程的一部分是确定 iSCSI 目标搜索方法，提供了具有可用目标列表的启动器。您可以通过静态、SendTargets 或 iSNS 动态搜索来配置 iSCSI 目标。使用 SendTargets 选项的动态搜索是访问大量目标的 iSCSI 启动器的最佳配置，如通过 iSCSI 到光纤通道网桥。SendTargets 动态搜索需要 iSCSI 目标的 IP 地址和端口组合，以使 iSCSI 启动器执行目标搜索。最通用的搜索方法是 SendTargets。

配置目标搜索方法时，必须提供以下信息，这取决于选择哪一种方法：

- SendTargets – 目标 IP 地址
- iSNS – iSNS 服务器地址
- 静态 – 目标 IP 地址和目标名称

有关配置目标搜索方法的更多信息，请参见[“配置动态或静态目标搜索” \[50\]](#)。

1. 启用 iSCSI 启动器服务。

```
initiator# svcadm enable network/iscsi/initiator
```

2. 登录到提供目标的服务器时，验证目标名称和 IP 地址。

```
target# ipadm show-addr
ADDROBJ      TYPE      STATE      ADDR
lo0/v4        static    ok         127.0.0.1/8
e1000g0/_b    dhcp      ok         1.2.3.4/24
lo0/v6        static    ok         ::1/128
e1000g0/_a    addrconf ok         fe80::123:1234:fe27:360c/10

target# itadm list-target -v
TARGET NAME                                     STATE  SESSIONS
iqn.1986-03.com.sun:02:73d12edc-9bb9-cb44-efc4-c3b36c123456  online    0
  alias:                                         -
  auth:                                         none (defaults)
  targetchapuser:                             -
  targetchapsecret:                           unset
  tpg-tags:                                    default
```

3. 配置要静态搜索的目标。

```
initiator# iscsiadm add static-config iqn.1986-03.com.sun:02:73d12edc-9bb9-cb44-efc4-c3b36c123456,
```

1.2.3.4

4. 查看静态配置信息。

```
initiator# iscsiadm list static-config
Static Configuration Target: iqn.1986-03.com.sun:02:73d12edc-9bb9-cb44-efc4-c3b36c123456,
1.2.3.4:3260
```

启用搜索方法之前，不启动 iSCSI 连接。请参见下一步。

5. 配置以下目标搜索方法之一：

- 如果已配置动态搜索 (SendTargets) 目标，请配置 SendTargets 搜索方法。

```
initiator# iscsiadm add discovery-address 1.2.3.4
```

- 如果已配置动态搜索 (iSNS) 目标，请配置 iSNS 搜索方法。

```
initiator# iscsiadm add isns-server 1.2.3.4
```

6. 启用以下目标搜索方法之一：

- 如果已配置动态搜索 (SendTargets) 目标，请启用 SendTargets 搜索方法。

```
initiator# iscsiadm modify discovery --sendtargets enable
```

- 如果已配置动态搜索 (iSNS) 目标，请启用 iSNS 搜索方法。

```
initiator# iscsiadm modify discovery --iSNS enable
```

- 如果已配置静态目标，请启用静态目标搜索方法。

```
initiator# iscsiadm modify discovery --static enable
```

7. 如有必要，请重新配置 /dev 名称空间来识别 iSCSI 磁盘。

```
initiator# devfsadm -i iscsi
```

▼ 如何访问 iSCSI 磁盘

Oracle Solaris iSCSI 启动器搜索到设备后，即会自动进行登录协商。Oracle Solaris iSCSI 驱动程序确定可用的 LU 数量并创建设备节点。然后，即可将 iSCSI 设备作为其他任何 SCSI 设备对待。

可以在 LU 上创建 ZFS 存储池，然后创建 ZFS 文件系统。

可以使用 `format` 实用程序查看本地系统上的 iSCSI 磁盘。

1. 在 `format` 输出中查看 iSCSI LU 信息。

```
initiator# format
```

```

0. c0t600144F0B5418B0000004DDAC7C10001d0 <SUN-COMSTAR-1.0 cyl 1022 alt 2 hd 128 sec 32>
   /scsi_vhci/disk@g600144f0b5418b00000004ddac7c10001
1. c8t0d0 <Sun-STK RAID INT-V1.0 cyl 17830 alt 2 hd 255 sec 63>
   /pci@0,0/pci10de,375@f/pci108e,286@0/disk@0,0
2. c8t1d0 <Sun-STK RAID INT-V1.0-136.61GB>
   /pci@0,0/pci10de,375@f/pci108e,286@0/disk@1,0
3. c8t2d0 <Sun-STK RAID INT-V1.0-136.61GB>
   /pci@0,0/pci10de,375@f/pci108e,286@0/disk@2,0
4. c8t3d0 <Sun-STK RAID INT-V1.0 cyl 17830 alt 2 hd 255 sec 63>
   /pci@0,0/pci10de,375@f/pci108e,286@0/disk@3,0
Specify disk (enter its number): 0
selecting c0t600144F0B5418B0000004DDAC7C10001d0
[disk formatted]

```

在以上输出中，磁盘 0 是受 MPxIO 控制的 iSCSI LU。磁盘 1-4 为本地磁盘。

2. 可以在 iSCSI LU 上创建 ZFS 存储池和 ZFS 文件系统。

```

initiator# zpool create pool-name c0t600144F0B5418B0000004DDAC7C10001d0
initiator# zfs create pool-name/fs-name

```

ZFS 文件系统在创建时会自动挂载，并在引导时重新挂载。

▼ 如何删除搜索到的目标设备

执行以下任一操作之后，将注销关联目标：

- 删除搜索地址
- 删除 iSNS 服务器
- 删除静态配置
- 禁用搜索方法

如果这些关联目标仍在（例如，已挂载文件系统），则注销这些设备将失败，设备仍然位于活动目标列表中。

此可选过程假定您已登录到本地系统，且此系统上已配置了对目标设备的访问权限。

1. 成为管理员。
2. （可选）使用以下操作之一禁用 iSCSI 目标搜索方法：
 - 如果要禁用 SendTargets 搜索方法，请使用以下命令：

```
initiator# iscsiadm modify discovery --sendtargets disable
```

- 如果要禁用 iSNS 搜索方法，请使用以下命令：

```
initiator# iscsiadm modify discovery --iSNS disable
```

- 如果要禁用静态目标搜索方法，请使用以下命令：

```
initiator# iscsiadm modify discovery --static disable
```

3. 使用以下操作之一删除 iSCSI 设备搜索项：

- 删除 iSCSI SendTargets 搜索项。

例如：

```
initiator# iscsiadm remove discovery-address 1.2.3.4:3260
```

- 删除 iSCSI iSNS 搜索项。

例如：

```
# iscsiadm remove isns-server 1.2.3.4:3205
```

- 删除静态 iSCSI 搜索项。

例如：

```
initiator# iscsiadm remove static-config eui.5000ABCD78945E2B,1.2.3.4
```

注 - 如果尝试禁用或删除其关联逻辑单元 (Logical Unit, LU) 正在使用的搜索项，则禁用或删除操作将失败，并会显示以下消息：

```
logical unit in use
```

如果出现这种错误，请停止 LU 中的所有关联 I/O，卸载文件系统。然后，重复执行禁用或删除操作。

在基于 iSCSI 的存储网络中配置验证

为 iSCSI 设备设置验证是可选的。

在安全环境下，由于仅有受信任的启动器才能访问目标，因此不需要进行验证。

在安全性较低的环境中，目标不能确定连接请求是否真正来自给定主机。在这种情况下，目标可以使用质询握手身份验证协议 (Challenge-Handshake Authentication Protocol, CHAP) 来验证启动器。

CHAP 验证使用质询和响应的概念，这意味着目标将质询启动器以查验其身份。要使质询/响应方法可行，目标必须知道启动器的密钥，并且必须将启动器设置为响应质询。有关在数组中设置密钥的说明，请参阅数组供应商的文档。

iSCSI 支持单向和双向验证，如下：

- 单向验证使目标可以验证启动器的身份。单向验证代表目标验证启动器。
- 双向验证通过使启动器可以验证目标身份，提供了更高级别的安全性。双向验证由启动器驱动，控制是否执行了双向验证。该目标唯一所需的设置是必须正确定义 chap 用户和 chap 机密。

▼ 如何配置对 iSCSI 启动器的 CHAP 验证

此过程假定您已登录到本地系统，并且要在此系统中安全地访问已配置的 iSCSI 目标设备。

- COMSTAR iSCSI 目标的 CHAP 密钥长度必须最少为 12 个字符，最多为 255 个字符。有些启动器只支持少于最大长度的密钥。
- 使用 CHAP 识别自己的每个节点必须具有用户名和口令。缺省情况下，在 Oracle Solaris OS 中会将 CHAP 用户名设置为启动器或目标节点名称（即 iqn 名称）。可以将 CHAP 用户名设置为小于 512 个字节的任意长度的文本。512 个字节长度限制是 Oracle Solaris 的限制。但是，如果未设置 CHAP 用户名，则系统会在初始化时将其设置为节点名称。

可以使用第三方 RADIUS 服务器来简化 CHAP 密钥管理，RADIUS 服务器充当集中验证服务。使用 RADIUS 时，RADIUS 服务器会存储节点名称组并匹配 CHAP 密钥。执行验证的系统将请求程序的节点名称和提供的机密转发给 RADIUS 服务器。RADIUS 服务器确认密钥是否是验证给定节点名称的合适密钥。iSCSI 和 iSER 均支持使用 RADIUS 服务器。

有关使用第三方 RADIUS 服务器的更多信息，请参见[“使用第三方 RADIUS 服务器简化 iSCSI 配置中的 CHAP 管理” \[57\]](#)。

1. 成为管理员。

有关更多信息，请参见《[在 Oracle Solaris 11.2 中确保用户和进程的安全](#)》中的[“使用所指定的管理权限”](#)。

2. 确定要配置单向 CHAP 还是双向 CHAP。

- 单向验证可使目标验证启动器。此方法为缺省方法。只需完成步骤 3-5 即可。
- 双向验证通过使启动器验证目标提供了更高级别的安全性。请完成步骤 3-9。

3. 单向 CHAP：设置启动器的密钥。

以下命令将启动一个用于定义 CHAP 密钥的对话框：

```
initiator# iscsiadm modify initiator-node --CHAP-secret
Enter CHAP secret: *****
Re-enter secret: *****
```

4. （可选）单向 CHAP：设置启动器的 CHAP 用户名。

缺省情况下，启动器的 CHAP 用户名会设置为启动器节点名称。

使用以下命令来使用启动器的 CHAP 用户名：

```
initiator# iscsiadm modify initiator-node --CHAP-name new-CHAP-name
```

5. 单向 CHAP – 在启动器上启用 CHAP 验证。

```
initiator# iscsiadm modify initiator-node --authentication CHAP
```

CHAP 要求启动器节点具有用户名和口令。用户名通常由目标用于查找给定用户名的密钥。

6. 选择以下步骤之一启用或禁用双向 CHAP。

- 启用双向 CHAP 以与目标连接。

```
initiator# iscsiadm modify target-param -B enable target-iqn
```

- 禁用双向 CHAP。

```
initiator# iscsiadm modify target-param -B disable target-iqn
```

7. 双向 CHAP : 针对目标将验证方法设置为 CHAP。

```
initiator# iscsiadm modify target-param --authentication CHAP target-iqn
```

8. 双向 CHAP : 设置确定目标的目标设备密钥。

以下命令将启动一个用于定义 CHAP 密钥的对话框：

```
initiator# iscsiadm modify target-param --CHAP-secret target-iqn
```

9. 双向 CHAP : 如果目标使用备用 CHAP 用户名，请设置确定目标的 CHAP 名称。
缺省情况下，目标的 CHAP 名称会设置为目标名称。

可以使用以下命令来更改目标的 CHAP 名称：

```
initiator# iscsiadm modify target-param --CHAP-name target-CHAP-name
```

▼ 如何配置对 iSCSI 目标的 CHAP 验证

此过程假定您已登录到包含 iSCSI 目标的本地系统。

1. 成为管理员。
2. 确定要配置单向 CHAP 还是双向 CHAP。
 - 单向验证为缺省方法。只需完成步骤 3-5 即可。
 - 对于双向验证，请完成步骤 3-7。
3. 单向/双向 CHAP : 配置目标来要求启动器确定正在使用 CHAP。

```
target# itadm modify-target -a chap target-iqn
```

4. 单向/双向 CHAP : 创建描述启动器的启动器上下文。

使用启动器的完整节点名称和启动器的 CHAP 密钥创建启动器上下文。

```
target# itadm create-initiator -s initiator-iqn
Enter CHAP secret: *****
Re-enter secret: *****
```

5. 单向/双向 CHAP : 如果启动器使用备用 CHAP 名称, 则使用备用名称配置启动器上下文。

```
target# itadm modify-initiator -u initiator-CHAP-name initiator-iqn
```

6. 双向 CHAP : 设置确定目标的目标设备密钥。

```
target# itadm modify-target -s target-iqn
Enter CHAP secret: *****
Re-enter secret: *****
```

7. (可选) 双向 CHAP : 如果目标使用备用 CHAP 用户名而不是目标节点名称 (iqn), 请修改目标。

```
target# itadm modify-target -u target-CHAP-name target-iqn
```

使用第三方 RADIUS 服务器简化 iSCSI 配置中的 CHAP 管理

可以使用第三方 RADIUS 服务器来简化 CHAP 密钥管理。RADIUS 服务器充当集中验证服务。使用此方法时, 推荐的做法是使用每个启动器节点的缺省 CHAP 名称。通常情况下, 所有启动器使用缺省 CHAP 名称时, 不必在目标上创建启动器上下文。

▼ 如何为 iSCSI 目标配置 RADIUS 服务器

可以使用第三方 RADIUS 服务器来简化 CHAP 密钥管理。RADIUS 服务器充当集中验证服务。使用此方法时, 推荐的做法是使用每个启动器节点的缺省 CHAP 名称。通常情况下, 所有启动器使用缺省 CHAP 名称时, 不必在目标上创建启动器上下文。

此过程假定您已登录到本地系统, 并且要在此系统中安全地访问已配置的 iSCSI 目标设备。

1. 成为管理员。
2. 使用 RADIUS 服务器的 IP 地址和端口配置启动器节点。
缺省端口为 1812。一次在目标系统上针对所有 iSCSI 目标完成该配置。

```
initiator# itadm modify-defaults -r RADIUS-server-IP-address
Enter RADIUS secret: *****
```

```
Re-enter secret: *****
```

3. 配置用于目标系统和 RADIUS 服务器之间通信的共享密钥。

```
initiator# itadm modify-defaults -d
Enter RADIUS secret: *****
Re-enter secret: *****
```

4. 配置目标系统来要求 RADIUS 验证。

可以为单个目标执行该配置，或作为所有目标的缺省配置。

```
initiator# itadm modify-target -a radius target-iqn
```

5. 使用以下组件配置 RADIUS 服务器：

- 目标节点的标识（例如其 IP 地址）
- 目标节点用来与 RADIUS 服务器通信的共享密钥
- 每个启动器需要验证的启动器 CHAP 名称（例如其 iqn 名称）和密钥

▼ 如何为 iSCSI 启动器配置 RADIUS 服务器

可以使用第三方 RADIUS 服务器来简化 CHAP 密钥管理。RADIUS 服务器充当集中验证服务。此步骤只在启动器要求双向 CHAP 验证时才有用。尽管仍然必须指定启动器的 CHAP 密钥，但是，在将双向验证与 RADIUS 服务器结合使用时，不再需要在启动器上指定每个目标的 CHAP 密钥。RADIUS 可在启动器或目标上独立配置。启动器和目标不必使用 RADIUS。

1. 成为管理员。
2. 使用 RADIUS 服务器的 IP 地址和端口配置启动器节点。
缺省端口为 1812。

```
# iscsiadm modify initiator-node --radius-server ip-address:1812
```

3. 使用 RADIUS 服务器的共享密钥配置启动器节点。
iSCSI 与 RADIUS 服务器进行交互之前必须使用共享密钥配置 RADIUS 服务器。

```
# iscsiadm modify initiator-node --radius-shared-secret
Enter secret:
Re-enter secret
```

4. 启用 RADIUS 服务器。

```
# iscsiadm modify initiator-node --radius-access enable
```

5. 设置 CHAP 双向验证的其他方面。

```
# iscsiadm modify initiator-node --authentication CHAP
# iscsiadm modify target-param --bi-directional-authentication enable target-iqn
# iscsiadm modify target-param --authentication CHAP target-iqn
```

6. 使用以下组件配置 RADIUS 服务器：

- 此节点的标识（例如其 IP 地址）
- 此节点用来与 RADIUS 服务器通信的共享密钥
- 每个需要验证的目标的 CHAP 名称（例如其 iqn 名称）和密钥

Oracle Solaris iSCSI 和 RADIUS 服务器错误消息

本节介绍与 Oracle Solaris iSCSI 和 RADIUS 服务器配置相关的错误消息。还介绍了可能的恢复解决方法。

empty RADIUS shared secret

原因: 已在启动器上启用了 RADIUS 服务器，但未设置 RADIUS 共享密钥。

解决方法: 使用 RADIUS 共享密钥配置启动器。有关更多信息，请参见[如何为 iSCSI 目标配置 RADIUS 服务器 \[57\]](#)。

WARNING: RADIUS packet authentication failed

原因: 启动器无法验证 RADIUS 数据包。如果在启动器节点上配置的共享密钥不同于 RADIUS 服务器上的共享密钥，则会出现此错误。

解决方法: 使用正确的 RADIUS 共享密钥重新配置启动器。有关更多信息，请参见[如何为 iSCSI 目标配置 RADIUS 服务器 \[57\]](#)。

设置 Oracle Solaris iSCSI 多路径设备

多路径 I/O (MPxIO) 使得 I/O 设备可通过 I/O 设备单个实例中的多个主机控制器接口接受访问。

使用 Oracle Solaris iSCSI 多路径 (MPxIO) 设备时考虑以下原则：

- Oracle Solaris iSCSI 和 MPxIO – MPxIO 支持 Oracle Solaris iSCSI 配置中目标端口聚合和可用性，该配置在 iSCSI 启动器上为每个目标 (MS/T) 配置多个会话。
 - 为两个或更多 NIC 聚合和故障转移使用 IP 网络多路径 (IPMP)。

- iSCSI 主机的基本配置是一个带有两个专用于 iSCSI 流量的 NIC 的服务器。使用 IPMP 配置 NIC。为非 iSCSI 流量提供附加的 NIC 以优化性能。
- 只能通过使用 Oracle Solaris iSCSI MS/T 功能实现活动的多路径，以及 IPMP 配置的故障转移和冗余。
 - 如果一个 NIC 在 IPMP 配置中出现故障，则 IPMP 处理故障转移。MPxIO 驱动程序不检测故障。在非 IPMP 配置中，MPxIO 驱动程序出现故障并脱机。
 - 如果一个目标端口在 IPMP 配置中出现故障，则 MPxIO 驱动程序检测故障并提供故障转移。在非 IPMP 配置中，MPxIO 驱动程序检测故障并提供故障转移。

有关为每个目标配置多个会话的更多信息，请参见[如何为目标启用多个 iSCSI 会话 \[60\]](#)。有关配置 IPMP 的信息，请参见《[在 Oracle Solaris 11.2 中管理 TCP/IP 网络、IPMP 和 IP 隧道](#)》中的第 3 章“管理 IPMP”。

- Oracle Solaris iSCSI、光纤通道 (Fibre Channel, FC) 和 MPxIO – MPxIO 驱动程序在更复杂的 iSCSI/FC 配置中提供以下行为：
 - 如果在 FC SAN 中具有到 FC 网桥的双 iSCSI，iSCSI 显示到 MPxIO 的目标路径。MPxIO 与每个 LU 标识符的唯一 SCSI 相匹配，如果它们相同则显示指向 iSCSI 驱动程序的一个路径。
 - 如果具有使用 iSCSI 和 FC 连接目标的配置，MPxIO 驱动程序可提供到相同设备的不同传输。在此配置中，MPxIO 使用这两种路径。
 - 如果与 MPxIO 一同使用 iSCSI 和 FC，确保在 `/etc/driver/drv/fp.conf` 和 `/driver/drv/iscsi.conf` 文件中的 MPxIO 参数与要支持的 MPxIO 配置相匹配。例如，在 `fp.conf` 中，可以确定 MPxIO 是否在 HBA 或每个端口的基础上全局启用。
- 第三方硬件考虑因素 – 找出第三方 HBA 是否有资格与 Oracle Solaris iSCSI 和 MPxIO 协同工作。

如果正在使用第三方 HBA，可能需要询问供应商有关 `/driver/drv/scsi_vhci.conf` 文件的对称选项信息。

▼ 如何为目标启用多个 iSCSI 会话

可以使用此过程创建与单个目标连接的多个 iSCSI 会话。此方案对于支持登录重定向或在同一个目标门户组中具有多个目标门户的 iSCSI 目标设备非常有用。应该将每个目标的多个 iSCSI 会话与 Oracle Solaris 的 SCSI 多路径 (MPxIO) 功能结合使用。如果在主机端使用多个 NIC 连接到同一目标的多个门户上，可以实现更高的带宽。

MS/T 功能通过改变启动器的会话 ID (ISID) 在目标上创建两个或更多会话。启用此功能在网络上创建两个 SCSI 层路径，以便使多个目标通过 iSCSI 层暴露于 Oracle Solaris I/O 层。MPxIO 驱动程序处理这些路径上的预留空间。

有关 iSCSI 如何与 MPxIO 路径交互的更多信息，请参见[“设置 Oracle Solaris iSCSI 多路径设备” \[59\]](#)。

为 iSCSI 目标配置多个会话之前，请查看以下各项：

- 典型的 MS/T 配置具有两个或更多已配置的会话。
不过，如果存储支持多个 TPGT 而且正在主机系统上使用 SendTarget 搜索方法，则已配置的会话数可设置为 1。SendTarget 搜索自动检测是否存在多个路径，并创建多个目标会话。
- 确认 mpxio 配置参数已在 /etc/driver/drv/iscsi.conf 文件中启用。

```
# cd /etc/driver/drv
# grep mpxio iscsi.conf iscsi.conf
iscsi.conf:mpxio-disable="no";
```

- 确认已使用 IPMP 配置多个网络连接。
- 确认多个网络连接可用。

```
# ipadm show-addr
```

1. 成为管理员。
2. 列出 iSCSI 启动器和目标的当前参数值。

- a. 列出 iSCSI 启动器的当前参数值。

```
initiator# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zzr1200
.
.
.
Configured Sessions: 1
```

- b. 列出 iSCSI 目标设备的当前参数值。

```
initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
Alias: -
.
.
.
Configured Sessions: 1
```

已配置的会话值是为目标门户组中的每个目标名称创建的已配置 iSCSI 会话的数目。

3. 选择以下命令之一，修改启动器节点处已配置会话的数目，以便应用于所有目标；或修改某个目标级别的已配置会话的数目，以便应用于特定目标：

- 将所需的参数值应用于 iSCSI 启动器节点。

```
initiator# iscsiadm modify initiator-node -c 2
```

- 将所需的参数值应用于 iSCSI 目标。

```
initiator# iscsiadm modify target-param -c 2 iqn.1992-08.com.abcstorage:sn.84186266
```

- 将已配置的会话绑定到一个或多个本地 IP 地址。

还可以将已配置的会话绑定到一个本地 IP 地址。使用此方法时，在以逗号分隔的列表中将提供一个或多个本地 IP 地址。每个 IP 地址都表示一个 iSCSI 会话。此方法也可以在 initiator-node 或 target-param 级别使用。例如：

```
initiator# iscsiadm modify initiator-node -c 10.0.0.1,10.0.0.2
```

注 - 如果指定的 IP 地址无法路由，则将忽略该地址，并对此会话使用缺省的 Oracle Solaris 路由和 IP 地址。

4. 验证是否已修改参数。

- a. 显示更新后的启动器节点信息。

```
initiator# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zzr1200
.
.
.
Configured Sessions: 2
```

- b. 显示更新后的目标节点信息。

```
initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
Alias: -
.
.
.
Configured Sessions: 2
```

5. 列出多个路径以确认 OS 设备名称与 iscsiadm list 输出相匹配，并且路径计数为 2 或更多。

```
initiator# mpathadm list lu
```

监视 iSCSI 配置

通过使用 iscsiadm list 命令，可以显示有关 iSCSI 启动器和目标设备的信息。

▼ 如何显示 iSCSI 配置信息

1. 成为管理员。
2. 显示有关 iSCSI 启动器的信息。

例如：

```
# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zzrl200
    Login Parameters (Default/Configured):
        Header Digest: NONE/-
        Data Digest: NONE/-
    Authentication Type: NONE
    RADIUS Server: NONE
    RADIUS access: unknown
    Configured Sessions: 1
```

3. 显示有关当前使用的搜索方法的信息。

例如：

```
# iscsiadm list discovery
Discovery:
    Static: enabled
    Send Targets: enabled
    iSNS: enabled
```

例 4-1 显示 iSCSI 目标信息

以下示例说明如何显示特定 iSCSI 目标的参数值：

```
# iscsiadm list target-param iqn.1992-08.com.abcstorage:sn.33592219
Target: iqn.1992-08.com.abcstorage:sn.33592219
```

iscsiadm list target-param -v 命令显示以下信息：

- 目标的验证值
- 目标登录参数的缺省值
- 每个登录参数的已配置值

iscsiadm list target-param -v 命令在 / 指示符前显示缺省参数值，并在 / 指示符后显示已配置参数值。如果未配置任何参数，其值将显示为连字符 (-)。有关更多信息，请参见以下示例。

```
# iscsiadm list target-param -v eui.50060e8004275511 Target: eui.50060e8004275511
Alias: -
Bi-directional Authentication: disabled
Authentication Type: NONE
```

```
Login Parameters (Default/Configured):
  Data Sequence In Order: yes/-
  Data PDU In Order: yes/-
  Default Time To Retain: 20/-
  Default Time To Wait: 2/-
  Error Recovery Level: 0/-
  First Burst Length: 65536/-
  Immediate Data: yes/-
  Initial Ready To Transfer (R2T): yes/-
  Max Burst Length: 262144/-
  Max Outstanding R2T: 1/-
  Max Receive Data Segment Length: 65536/-
  Max Connections: 1/-
  Header Digest: NONE/-
  Data Digest: NONE/-
Configured Sessions: 1
```

以下示例输出显示了目标与启动器之间协商的参数：

```
# iscsiadm list target -v eui.50060e8004275511
Target: eui.50060e8004275511
  TPGT: 1
  ISID: 4000002a0000
  Connections: 1
    CID: 0
      IP address (Local): 172.20.101.71:32813
      IP address (Peer): 172.20.101.40:3260
      Discovery Method: Static
      Login Parameters (Negotiated):
        Data Sequence In Order: yes
        Data PDU In Order: yes
        Default Time To Retain: 0
        Default Time To Wait: 3
        Error Recovery Level: 0
        First Burst Length: 65536
        Immediate Data: yes
        Initial Ready To Transfer (R2T): yes
        Max Burst Length: 262144
        Max Outstanding R2T: 1
        Max Receive Data Segment Length: 65536
        Max Connections: 1
        Header Digest: NONE
        Data Digest: NONE
```

修改 iSCSI 启动器和目标参数

可以同时修改 iSCSI 启动器和 iSCSI 目标设备上的参数。但是，只能在 iSCSI 启动器上修改的参数包括以下各项：

- iSCSI 启动器节点名称 – 可以将启动器节点名称更改为不同的名称。如果更改了启动器节点名称，则系统可能会从启动器的目标列表中删除 iSNS 搜索到的目标，具体

取决于更改名称时 iSNS 服务器上的搜索域配置。有关更多信息，请参见[如何修改 iSCSI 启动器和目标参数 \[67\]](#)。

- 头摘要 – NONE（缺省值），或者 CRC32。
- 数据摘要 – NONE（缺省值），或者 CRC32。
- 验证和 CHAP 密钥 – 有关设置验证的更多信息，请参见[如何配置对 iSCSI 启动器的 CHAP 验证 \[55\]](#)。
- 已配置的会话 – 有关配置多个会话的更多信息，请参见[如何为目标启用多个 iSCSI 会话 \[60\]](#)。

iSCSI 驱动程序为 iSCSI 启动器和 iSCSI 目标设备参数提供缺省值。如果修改了 iSCSI 启动器的参数，则修改后的参数将由 iSCSI 目标设备继承，除非 iSCSI 目标设备已有不同的值。



注意 - 请确保目标软件支持要修改的参数。否则，可能无法登录到 iSCSI 目标设备。有关支持的参数的列表，请参见数组文档。

应在启动器与目标之间的 I/O 完成后再修改 iSCSI 参数。使用 `iscsiadm modify` 命令进行更改后，iSCSI 驱动程序将重新连接会话。

调优 iSCSI 参数

可以调优 iSCSI 参数以适应 iSCSI 启动器的各种响应或连接时间值。可以调优 iSCSI 参数，取决于是否要在启动器上为所有目标调整参数（其中启动器系统已连接），或者为一个特定目标调整参数。

使用以下命令更改特定 iSCSI 目标的参数值。

```
iscsiadm modify target-param -T tunable-prop=value target-name
```

使用以下命令调整所有目标的参数值：

```
iscsiadm modify initiator-node -T tunable-prop=value
```

以下可调参数应用于活动连接，并更改 iSCSI 启动器和连接到启动器的目标的行为。配置 iSCSI 启动器时，动态调优参数的能力提供了灵活性。

表 4-1 iSCSI 可调参数

参数名称	说明	有效值（秒）	缺省值（秒）
recv-login-rsp-timeout	会话登录响应时间 – 指定 iSCSI 启动器等待来自给定 iSCSI 目标的 iSCSI 会话登录请求的响应时间。	0 – 3600	60
conn-login-max	最大连接重试时间 – iSCSI 启动器连接到目标 I/O 超时或连接失败之后，确定	0 – 3600	180

参数名称	说明	有效值 (秒)	缺省值 (秒)
	iSCSI 启动器尝试连接到目标的最大次数。		
polling-login-delay	登录重试时间间隔 – iSCSI 启动器连接到目标 I/O 超时或连接失败之后，确定每个 iSCSI 会话登录重试之间的时间间隔。	0 – 3600	60

▼ 如何调优 iSCSI 参数

1. 显示所有可调 iSCSI 参数。
显示所有目标的 iSCSI 参数信息。

```
# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:e00000000000.4e36d278
Initiator node alias: unknown
  Login Parameters (Default/Configured):
    Header Digest: NONE/-
    Data Digest: NONE/-
    Max Connections: 65535/-
  Authentication Type: NONE
  RADIUS Server: NONE
  RADIUS Access: disabled
  Tunable Parameters (Default/Configured):
    Session Login Response Time: 60/-
    Maximum Connection Retry Time: 180/-
    Login Retry Time Interval: 60/-
  Configured Sessions: 1
```

显示特定目标的 iSCSI 参数信息。

```
# iscsiadm list target-param [target-name]
```

例如：

```
# iscsiadm list target-param -v iqn.1986-03.com.sun:02:47ac0
506-cd48-67f5-fc0d-ab7544d37538
Target: iqn.1986-03.com.sun:02:47ac0506-cd48-67f5-fc0d-ab7544d37538
Alias: -
Bi-directional Authentication: disabled
Authentication Type: NONE
  Login Parameters (Default/Configured):
    Data Sequence In Order: yes/-
    Data PDU In Order: yes/-
    Default Time To Retain: 20/-
    Default Time To Wait: 2/-
    Error Recovery Level: 0/-
    First Burst Length: 65536/-
    Immediate Data: yes/-
    Initial Ready To Transfer (R2T): yes/-
    Max Burst Length: 262144/-
    Max Outstanding R2T: 1/-
```

```

Max Receive Data Segment Length: 8192/-
Max Connections: 65535/-
Header Digest: NONE/-
Data Digest: NONE/-
Tunable Parameters (Default/Configured):
Session Login Response Time: 60/-
Maximum Connection Retry Time: 180/-
Login Retry Time Interval: 60/-
Configured Sessions: 1

```

2. 调优 iSCSI 参数。

例如，将一个目标的最大连接重试时间设为 90 秒：

```
# iscsiadm modify target-param -T conn-login-max=90 iqn.1986-03.com.sun:02:47ac0506-cd48-67f5-fc0d-ab7544d37538
```

例如，将所有目标的最大连接重试时间设为 90 秒：

```
# iscsiadm modify initiator-node -T conn-login-max=90
```

▼ 如何修改 iSCSI 启动器和目标参数

此过程的第一部分说明了 iSCSI 目标设备如何继承 iSCSI 启动器的修改参数。此过程的第二部分说明如何在 iSCSI 目标设备上实际修改参数。

此可选过程假定您已登录到本地系统，在此系统上已配置了对 iSCSI 目标设备的访问权限。

1. 成为管理员。
2. 列出 iSCSI 启动器和目标设备的当前参数值。

a. 列出 iSCSI 启动器的当前参数值。

```

initiator# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zzr1200
Login Parameters (Default/Configured):
Header Digest: NONE/-
Data Digest: NONE/-
Authentication Type: NONE
RADIUS Server: NONE
RADIUS access: unknown
Configured Sessions: 1

```

b. 列出 iSCSI 目标设备的当前参数值。

```

initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266

```

```
Alias: -
Bi-directional Authentication: disabled
Authentication Type: NONE
Login Parameters (Default/Configured):
  Data Sequence In Order: yes/-
  Data PDU In Order: yes/-
  Default Time To Retain: 20/-
  Default Time To Wait: 2/-
  Error Recovery Level: 0/-
  First Burst Length: 65536/-
  Immediate Data: yes/-
  Initial Ready To Transfer (R2T): yes/-
  Max Burst Length: 262144/-
  Max Outstanding R2T: 1/-
  Max Receive Data Segment Length: 65536/-
  Max Connections: 1/-
  Header Digest: NONE/-
  Data Digest: NONE/-
Configured Sessions: 1
```

请注意，对于 iSCSI 启动器和 iSCSI 目标设备，当前已将头摘要和数据摘要参数设置为 NONE。

要查看 iSCSI 目标设备的缺省参数值，请参见[例 4-1 “显示 iSCSI 目标信息”](#)中的 `iscsiadm list target-param` 输出。

3. 修改 iSCSI 启动器的参数。

例如，将头摘要设置为 CRC32。

```
initiator# iscsiadm modify initiator-node -h CRC32
```

如果更改了启动器节点名称，则系统可能会注销 iSNS 搜索到的目标，并将其从启动器的目标列表中删除，前提是新名称与目标名称不属于同一搜索域。但是，如果当前正在使用目标，则不会将其删除。例如，如果某个文件已打开或已在这些目标上挂载了文件系统，将不会删除目标。

如果这些目标和新启动器节点名称属于同一搜索域，则在名称更改后还可能看到新目标。

4. 验证是否已修改参数。

a. 显示 iSCSI 启动器的更新后参数信息。

```
initiator# iscsiadm list initiator-node
Initiator node name: iqn.1986-03.com.sun:01:0003ba4d233b.425c293c
Initiator node alias: zzr1200
  Login Parameters (Default/Configured):
    Header Digest: NONE/CRC32
    Data Digest: NONE/-
  Authentication Type: NONE
  RADIUS Server: NONE
  RADIUS access: unknown
```

Configured Sessions: 1

请注意，现在已将头摘要设置为 CRC32。

b. 显示 iSCSI 目标设备的更新后参数信息。例如：

```
initiator# iscsiadm list target-param -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
Alias: -
Bi-directional Authentication: disabled
Authentication Type: NONE
Login Parameters (Default/Configured):
  Data Sequence In Order: yes/-
  Data PDU In Order: yes/-
  Default Time To Retain: 20/-
  Default Time To Wait: 2/-
  Error Recovery Level: 0/-
  First Burst Length: 65536/-
  Immediate Data: yes/-
  Initial Ready To Transfer (R2T): yes/-
  Max Burst Length: 262144/-
  Max Outstanding R2T: 1/-
  Max Receive Data Segment Length: 65536/-
  Max Connections: 1/-
  Header Digest: CRC32/-
  Data Digest: NONE/-
Configured Sessions: 1
```

请注意，现在已将头摘要设置为 CRC32。

5. 验证 iSCSI 启动器已重新连接至 iSCSI 目标。

```
initiator# iscsiadm list target -v iqn.1992-08.com.abcstorage:sn.84186266
Target: iqn.1992-08.com.abcstorage:sn.84186266
  TPGT: 2
  ISID: 4000002a0000
  Connections: 1
    CID: 0
      IP address (Local): nnn.nn.nn.nnn:64369
      IP address (Peer): nnn.nn.nn.nnn:3260
      Discovery Method: SendTargets
      Login Parameters (Negotiated):
        .
        .
        .
      Header Digest: CRC32
      Data Digest: NONE
```

6. (可选) 重置 iSCSI 启动器参数或 iSCSI 目标设备参数。

通过使用 `iscsiadm modify` 命令可以将参数重置为其缺省值。或者，可以使用 `iscsiadm remove` 命令将所有参数重置为其缺省值。

`iscsiadm modify target-param` 命令仅更改在命令行中指定的参数。

以下示例说明如何将头摘要重置为 NONE：

```
initiator# iscsiadm modify target-param -h none iqn.1992-08.com.abcstorage:sn...
```

有关 `iscsiadm remove target-param` 命令的信息，请参见[iscsiadm\(1M\)](#)。

iSCSI 配置问题的故障排除

以下工具可用于解决一般的 iSCSI 配置问题：

- `snoop` – 此工具已更新为支持 iSCSI 软件包。
- `wireshark` – 可以从 <http://www.wireshark.org/> 获取此产品。

这两个工具都可以筛选端口 3260 上的 iSCSI 数据包。

以下各节介绍了各种 iSCSI 故障排除和错误消息解析方案。

从本地系统到 iSCSI 目标无连接

▼ 如何解决 iSCSI 连接问题

1. 成为管理员。
2. 列出 iSCSI 目标信息。

例如：

```
initiator# iscsiadm list target
Target: iqn.2001-05.com.abcstorage:6-8a0900-37ad70401-bcfff02df8a421df-zzr1200-01
        TPGT: default
        ISID: 4000002a0000
        Connections: 0
```

3. 如果 `iscsiadm list target` 输出中未列出任何连接，请检查 `/var/adm/messages` 文件，以找出连接失败的可能原因。
通过使用 `ping` 命令或使用 `telnet` 命令与存储设备的 iSCSI 端口连接，还可验证连接是否可访问，从而可确保 iSCSI 服务可用。缺省端口为 3260。
此外，请检查存储设备的日志文件中是否有错误。
4. 如果目标未列在 `iscsiadm list target` 输出中，请检查 `/var/adm/messages` 文件，以找出可能的原因。
如果当前使用 `SendTargets` 作为搜索方法，请尝试使用 `-v` 选项列出 `discovery-address`，以确保预期目标对于主机是可见的。例如：

```
initiator# iscsiadm list discovery-address -v 10.0.0.1
Discovery Address: 10.0.0.1:3260
Target name: eui.210000203787dfc0
Target address: 10.0.0.1:11824
Target name: eui.210000203787e07b
Target address: 10.0.0.1:11824
```

如果当前使用 iSNS 作为搜索方法，请尝试启用 iSNS 搜索模式，并使用 -v 选项列出 *isns-server*，以确保预期目标对主机是可见的。例如：

```
initiator# iscsiadm list isns-server -v
iSNS Server IP Address: 10.20.56.56:3205
Target name: iqn.1992-08.com.xyz:sn.1234566
Target address: 10.20.57.161:3260, 1
Target name: iqn.2003-10.com.abc:group-0:154:abc-65-01
Target address: 10.20.56.206:3260, 1
Target name: iqn.2003-10.com.abc:group-0:154:abc-65-02
Target address: 10.20.56.206:3260, 1
.
.
.
```

iSCSI 设备或磁盘在本地系统上不可用

▼ 如何解决 iSCSI 设备或磁盘的不可用性问题

1. 成为管理员。
2. 确定在枚举过程中在该目标上搜索到的 LUN。
-s 选项可显示在枚举过程中在该目标上搜索到的 LUN。

例如：

```
# iscsiadm list target -S
Target: iqn.2001-05.com.abcstorage:6-8a0900-37ad70401-bcfff02df8a421df-zzr1200-01
TPGT: default
ISID: 4000002a0000
Connections: 1
LUN: 0
Vendor: ABCSTOR
Product: 0010
OS Device Name: /dev/rdisk/c3t34d0s2
```

3. 请查看 `/var/adm/messages` 文件，以查看是否报告了错误。
如果您认为某个 LUN 应该已列出，但却没有列出，请检查此日志文件。
4. 请检查存储设备的日志文件中是否有错误。

5. 请确保任何存储设备 LUN 屏蔽均已正确配置。

在使用 iSNS 搜索方法时使用 LUN 屏蔽

请避免将 iSNS 搜索域用作控制特定启动器的存储验证的方式。如果要确保只有获得授权的启动器才能访问 LUN，请改用 LUN 屏蔽。

如果在目标正在使用时从搜索域中删除该目标，则 iSCSI 启动器将不会从该目标注销。如果不需要该启动器访问该目标（以及关联的 LUN），则必须使用 LUN 屏蔽。仅从搜索域中删除目标是不够的。

一般 iSCSI 错误消息

本节介绍可能在 `/var/adm/messages` 文件中找到的 iSCSI 消息以及可能的恢复解决方法。

消息格式如下：

`iscsi TYPE (OID) STRING (STATUS-CLASS#/STATUS-DETAIL#)`

TYPE 连接或会话。

OID 连接或会话的对象 ID。此 ID 对于 OS 实例是唯一的。

STRING 条件的说明。

**STATUS-CLASS#/
STATUS-DETAIL#** 这些值将在 iSCSI 登录响应中返回，如 RFC 3720 所定义的那样。

`iscsi connection(OID) login failed - Miscellaneous iSCSI initiator errors.`

原因: 设备登录因某种形式的启动器错误而失败。

`iscsi connection(OID) login failed - Initiator could not be successfully authenticated.`

原因: 设备不能成功验证启动器。

解决方法: 如果适用，请验证 CHAP 名称、CHAP 口令或 RADIUS 服务器的设置是否正确。

`iscsi connection(OID) login failed - Initiator is not allowed access to the given target.`

原因: 该设备不允许启动器访问 iSCSI 目标设备。

解决方法: 验证启动器名称, 并确认该名称已由存储设备正确屏蔽或提供。

`iscsi connection(OID) login failed - Requested ITN does not exist at this address.`

原因: 该设备不提供对正在请求的 iSCSI 目标名称 (iSCSI target name, ITN) 的访问权限。

解决方法: 验证正确指定了启动器搜索信息, 并且正确配置了存储设备。

`iscsi connection(OID) login failed - Requested ITN has been removed and no forwarding address is provided.`

原因: 该设备无法再提供对正在请求的 iSCSI 目标名称 (iSCSI target name, ITN) 的访问权限。

解决方法: 验证正确指定了启动器搜索信息, 并且正确配置了存储设备。

`iscsi connection(OID) login failed - Requested iSCSI version range is not supported by the target.`

原因: 存储设备不支持启动器的 iSCSI 版本。

`iscsi connection(OID) login failed - No more connections can be accepted on this Session ID (SSID).`

原因: 存储设备无法接受此启动器节点到 iSCSI 目标设备的更多连接。

`iscsi connection(OID) login failed - Missing parameters (e.g., iSCSI initiator and/or target name).`

原因: 存储设备正在报告未正确指定启动器或目标名称。

解决方法: 请正确指定 iSCSI 启动器或目标名称。

`iscsi connection(OID) login failed - Target hardware or software error.`

原因: 存储设备遇到硬件或软件错误。

解决方法: 请查阅存储文档或与存储设备供应商联系以获得进一步的帮助。

`iscsi connection(OID) login failed - iSCSI service or target is not currently operational.`

原因: 存储设备当前无法运行。

解决方法: 请查阅存储文档或与存储设备供应商联系以获得进一步的帮助。

`iscsi connection(OID) login failed - Target has insufficient session, connection or other resources.`

原因: 存储设备具有的资源不足。

解决方法: 请查阅存储文档或与存储设备供应商联系以获得进一步的帮助。

`iscsi connection(OID) login failed - unable to initialize authentication`
`iscsi connection(OID) login failed - unable to set authentication`
`iscsi connection(OID) login failed - unable to set username`
`iscsi connection(OID) login failed - unable to set password`
`iscsi connection(OID) login failed - unable to set ipsec`
`iscsi connection(OID) login failed - unable to set remote authentication`

原因: 该启动器无法正确初始化或设置验证。

解决方法: 验证是否已正确配置启动器的验证设置。

`iscsi connection(OID) login failed - unable to make login pdu`

原因: 启动器无法根据启动器或存储设备的设置生成登录有效负荷数据单元 (payload data unit, PDU)。

解决方法: 请尝试重置所有目标登录参数或其他非缺省设置。

`iscsi connection(OID) login failed - failed to transfer login`
`iscsi connection(OID) login failed - failed to receive login response`

原因: 启动器无法通过网络连接传输或接收登录有效负荷数据单元 (payload data unit, PDU)。

解决方法: 验证网络连接是否可以访问。

`iscsi connection(OID) login failed - received invalid login response (OP CODE)`

原因: 存储设备采用非预期响应来响应登录。

`iscsi connection(OID) login failed - login failed to authenticate with target`

原因: 启动器无法验证存储设备。

解决方法: 验证是否已正确配置启动器的验证设置。

iscsi connection(*OID*) login failed - initiator name is required

原因: 必须配置启动器名称才能执行所有操作。

解决方法: 验证是否已配置启动器名称。

iscsi connection(*OID*) login failed - authentication receive failed
iscsi connection(*OID*) login failed - authentication transmit failed

原因: 该启动器无法传输或接收验证信息。

解决方法: 根据情况验证存储设备或 RADIUS 服务器的网络连接。

iscsi connection(*OID*) login failed - login redirection invalid

原因: 存储设备尝试将启动器重定向到无效目标。

解决方法: 请查阅存储文档或与存储设备供应商联系以获得进一步的帮助。

iscsi connection(*OID*) login failed - target protocol group tag mismatch, expected <TPGT>, received <TPGT>

原因: 启动器和目标的 TPGT (target portal group tag, 目标门户组标记) 不匹配。

解决方法: 请验证启动器或存储设备上的 TPGT 搜索设置。

iscsi connection(*OID*) login failed - can't accept *PARAMETER* in security stage

原因: 设备已在登录安全阶段使用不支持的登录参数进行了响应。

解决方法: 请记录参数名称以供参考。请查阅存储文档或与存储设备供应商联系以获得进一步的帮助。

iscsi connection(*OID*) login failed - HeaderDigest=CRC32 is required, can't accept *VALUE*

iscsi connection(*OID*) login failed - DataDigest=CRC32 is required, can't accept *VALUE*

原因: 对于此目标, 该启动器已配置为仅接受将 HeaderDigest 或 DataDigest 设置为 CRC32 的情况。设备返回值 *VALUE*。

解决方法: 验证启动器和设备摘要设置兼容。

iscsi connection(*OID*) login failed - HeaderDigest=None is required, can't accept *VALUE*

iscsi connection(*OID*) login failed - DataDigest=None is required, can't accept *VALUE*

原因: 对于此目标, 该启动器已配置为只接受将 HeaderDigest 或 DataDigest 设置为 NONE 的情况。设备返回值 *VALUE*。

解决方法: 验证启动器和设备摘要设置兼容。

iscsi connection(*OID*) login failed - can't accept *PARAMETER*

原因: 启动器不支持此参数。

iscsi connection(*OID*) login failed - can't accept MaxOutstandingR2T *VALUE*

原因: 该启动器不接受 MaxOutstandingR2T 值为所记录 *VALUE* 的情况。

iscsi connection(*OID*) login failed - can't accept MaxConnections *VALUE*

原因: 该启动器不接受最大连接数为所记录 *VALUE* 的情况。

iscsi connection(*OID*) login failed - can't accept ErrorRecoveryLevel *VALUE*

原因: 该启动器不接受错误恢复级别值为所记录 *VALUE* 的情况。

iscsi session(*OID*) *NAME* offline

原因: 此目标 *NAME* 的所有连接都已删除或失败。

iscsi connection(*OID*) failure - unable to schedule enumeration

原因: 启动器无法枚举此目标上的 LUN。

解决方法: 可以通过运行 `devfsadm -i iscsi` 命令来强制执行 LUN 枚举。有关更多信息, 请参见 [devfsadm\(1M\)](#)。

iscsi connection(*OID*) unable to connect to target *NAME* (errno:*ERRNO*)

原因: 启动器无法建立网络连接。

解决方法: 有关连接故障的特定 *ERRNO* 的信息, 请参见 `/usr/include/sys/errno.h` 文件。

配置虚拟光纤通道端口

本章提供用于配置 N 端口 ID 虚拟化 (N Port ID Virtualization, NPIV) 端口（也称为虚拟光纤通道端口）的步骤。

本章包含以下主题：

- “什么是 NPIV？” [77]
- “NPIV 的限制” [77]
- “使用 NPIV 端口” [78]

什么是 NPIV？

NPIV 是光纤通道工具，它允许一个光纤通道适配器有多个 N 端口 ID。每个 N 端口在 SAN 上都有唯一的标识（端口 WWN 和节点 WWN），可用于区域划分和 LUN 屏蔽 (LUN masking)。软区域划分是首选区域划分方法，可用于按端口 WWN 将端口分组。

NPIV 的限制

用于虚拟化光纤通道端口时，NPIV 有以下限制：

- NPIV 端口不能用于引导。
- NPIV 端口最好用于所含端口（虚拟或物理）相对较少的 SAN 中。此外，NPIV 能够创建大量端口，但有些目标可能没有足够的资源对其进行处理。存在此限制是因为如果 SAN 上的端口很多，则在 SAN 上处理状态更改通知 (state change notification, SCN) 需要花费大量时间。通过使用区域划分限制可见端口的数量，可解决大型 SAN 上的这一限制。
- MPxIO 可与 NPIV 一起使用，但您应该确保不同的路径在物理上是冗余的。
- 仅光纤拓扑中支持 NPIV，FC-AL 或点对点拓扑中不支持。
- 并非所有硬件都支持 NPIV。在 SAN 中，交换机和 HBA（尽管不是目标）都必须支持 NPIV。根据规范，HBA 最多应可支持 255 个虚拟端口，但此能力受到交换机上资源的限制。交换机可能必须更新到最新固件级别才能提供 NPIV 支持。

使用 NPIV 端口

通过使用 `fcadm` 命令，可为非虚拟化环境配置 NPIV。

`fcinfo` 和 `fcadm` 状态命令可用于确定 NPIV 端口的状态，不管这些端口是否通过 `fcadm` 创建。这两个命令还将报告物理端口与该端口上托管的虚拟端口之间的关系。

其他光纤通道命令（如 `luxadm` 和 `cfgadm`）报告 NPIV 信息，尽管虚拟端口与物理端口之间并无差别。

▼ 如何创建 NPIV 端口

开始之前 每个虚拟端口必须具有端口和节点名称。端口名称在 SAN 上必须唯一。您可以手动指定名称，也可以使用内置随机 WWN 生成器。如果您尝试注册重复的名称，大多数交换机将在新注册的 WWN 上报告错误状态，同时交换机不注册新 WWN。

有关可接受名称格式的更多信息，请参阅 T11 标准：Fibre Channel Framing and Signaling（光纤通道框架和信号）(FC-FS 2)。

如果您尝试在不支持 NPIV 的 HBA 上创建 NPIV 端口，将发生错误。如果您尝试在支持 NPIV 的 HBA 上创建 NPIV 端口，但该 HBA 与不支持 NPIV 的交换机相连，也将创建该端口，但状态为脱机。该状态将在 `fcinfo(1M)` 输出中报告。

1. 成为管理员。
2. 创建 NPIV 端口。

```
# fcdm create-npiv-port -p Virtual_Port_WWN -n Virtual_Node_WWN
PhysicalPort_port_WWN
```

如果不使用 `-p` 和 `-n` 选项，将会为虚拟端口和虚拟节点相应指定随机 WWN。

例 5-1 创建 NPIV 端口

以下示例将在 WWN 为 210000e08b170f1c 的物理 HBA 端口上创建一个 NPIV 端口，虚拟端口 WWN 设置为 2000000000000001，虚拟节点 WWN 设置为 2100000000000001。

```
# fcdm create-npiv-port -p 2000000000000001 -n 2100000000000001 210000e08b170f1c
```

▼ 如何删除 NPIV 端口

开始之前 可使用 `fcinfo hba-port` 命令显示 NPIV 端口的最新 WWN 值。

1. 成为管理员。
2. 删除 NPIV 端口。

```
# fcadm delete-npiv-port -p Virtual_Port_WWN -n Virtual_Node_WWN
PhysicalPort_port_WWN
```

例 5-2 删除 NPIV 端口

以下示例将删除物理 HBA 端口上 WWN 为 210000e08b170f1c 的 NPIV 端口。

```
# fcadm delete-npiv-port -p 2000000000000001 -n 2100000000000001 210000e08b170f1c
```

▼ 如何显示 NPIV 端口状态

1. 成为管理员。
2. 显示当前配置的 NPIV 端口。

```
# fcinfo hba-port
```

NPIV 端口列表将显示当前配置的 NPIV 端口的列表。

例 5-3 显示 NPIV 端口上的状态

以下示例显示 HBA 端口 210000e08b84f7eb 具有一个虚拟端口。

```
# fcinfo hba-port
HBA Port WWN: 210000e08b84f7eb
  Port Mode: Initiator
  Port ID: 10100
  OS Device Name: /dev/cfg/c7
  Manufacturer: QLogic Corp.
  Model: 375-3294-01
  Firmware Version: 04.04.00
  FCode/BIOS Version: BIOS: 1.4; fcode: 1.11; EFI: 1.0;
  Serial Number: 0402F00-0549112895
  Driver Name: qlc
  Driver Version: 20080430-0.00
  Type: N-port
  State: online
  Supported Speeds: 1Gb 2Gb 4Gb
  Current Speed: 4Gb
  Node WWN: 200000e08b84f7eb
  Max NPIV Ports: 63
  NPIV port list:
    Virtual Port1:
      Node WWN: 1110000000000000
      Port WWN: 1210000000000000
```


配置 FCoE 端口

本章介绍用于配置标准以太网接口上托管的 FCoE 端口的步骤。本章不适用于聚合网络适配器 (Converged Network Adapter, CNA) 上的硬件 FCoE 端口。

本章包含以下主题：

- “什么是 FCoE？” [81]
- “FCoE 的限制” [81]
- “配置 FCoE 端口” [82]

什么是 FCoE？

FCoE 是新的 T11 标准，用于通过增强以太网传输封装光纤通道帧。其目的是在数据中心启用网络聚合和具成本效益的 SAN 扩展。

FCoE 的限制

Solaris FCoE 启动器是通过软件实现的，目的在于使用标准以太网控制器。不过，您必须遵守 Solaris FCoE 实现的以下限制：

- FCoE 端口不能用于引导。
- 不能在 Oracle VM Server for SPARC 或 Oracle VM Server 3.0 for x86 客操作系统中配置 FCoE 端口。
- 光纤和点对点拓扑支持 FCoE。

并非所有硬件都支持 FCoE。FCoE 可与支持 802.3x PAUSE 和巨型帧且具有 GLDV3 驱动程序的以太网控制器结合使用。

配置 FCoE 端口

可通过使用 `fcadm` 命令配置 FCoE 端口。使用 `fcinfo` 和 `fcadm` 命令确定 FCoE 端口的状态。这两个命令还会报告以太网接口与该接口上托管的 FCoE 端口之间的关系。

其他光纤通道命令（如 `luxadm` 和 `cfgadm`）报告 FCoE 信息，尽管 FCoE 与本机 FC 端口之间并无差别。

▼ 如何创建 FCoE 端口

开始之前 在开始本过程之前，必须先执行以下任务：

- 在以太网接口上启用 802.3x（也称为 PAUSE）设置。此设置将确保无损以太网传输。
- 在以太网接口上启用巨型帧（大于 2.5 KB）。光纤通道数据帧可达到 2136 字节。
以太网硬件和驱动程序不同，这些设置也会有所不同。大多数情况下，您必须修改以太网接口的 `driver.conf` 文件，然后重新引导。有关如何启用这些功能的详细信息，请参见以太网接口的 `driver.conf` 文件。
每个虚拟端口必须具有端口和节点名称。端口名称在 SAN 上必须唯一。您可以手动指定名称，也可以使用内置 WWN 生成器。如果您尝试注册重复的名称，交换机将在新注册的 WWN 上报告错误状态，同时交换机不注册新 WWN。有关可接受名称格式的更多信息，请参阅 T11 标准：Fibre Channel Framing and Signaling（光纤通道框架和信号）(FC-FS 2)。
如果您尝试在不支持 FCoE 的网络接口上创建 FCoE 端口，将发生错误，同时不创建 FCoE 端口。
- 启用以下服务：

```
# svcadm enable svc:/system/fcoe_target:default
# svcadm enable svc:/system/stmf:default
```

1. 成为管理员。
2. 创建 FCoE 端口。

```
# fcdm create-fcoe-port -i -p Port_WWN -n Node_WWN Ethernet_Interface
```

如果所选以太网接口不支持多重单播地址，将提示您在该接口上明确启用混杂模式。

```
# fcdm create-fcoe-port -i -f Ethernet_Interface
```

例如：

```
# fcdm create-fcoe-port -i net0
```

▼ 如何删除 FCoE 端口

开始之前 可使用 `fcadm list-fcoe-ports` 命令显示托管 FCoE 端口的以太网接口。

1. 成为管理员。
2. 删除 FCoE 端口。

```
# fcdm delete-fcoe-port network_interface
```

例如：

```
# fcdm delete-fcoe-port net0
```

▼ 如何显示 FCoE 端口状态

1. 成为管理员。
2. 显示当前配置的 FCoE 端口的状态。

```
# fcinfo hba-port -e
```

例如：

```
# fcinfo hba-port -e
HBA Port WWN: 200000144fc1f5c8
  Port Mode: Initiator
  Port ID: 9a0042
  OS Device Name: /dev/cfg/c6
  Manufacturer: Sun Microsystems, Inc.
  Model: FCoE Virtual FC HBA
  Firmware Version: N/A
  FCode/BIOS Version: N/A
  Serial Number: N/A
  Driver Name: SunFC FCoEI v20090422-1.00
  Driver Version: v20090422-1.00
  Type: N-port
  State: online
  Supported Speeds: 1Gb 10Gb
  Current Speed: 10 Gb
  Node WWN: 100000144fc1f5c8
```

列出系统中所有 FCoE 端口的 FC 特定信息。

```
# fcdm list-fcoe-ports
```

例如：

```
# fcdm list-fcoe-ports
```

```
HBA Port WWN: 2000001b2165a630
  Port Mode: Initiator
  Port ID: e00033
  VLAN ID: 7
  Link Name: net2
  MTU Size: 2500
  Primary MAC Address: 00:1b:21:65:a6:30
  FCoE MAC Address: 0e:fc:00:e0:00:33
  Promiscuous Mode: Off
  State: Online
  FIP Mode: On
  FCoE Hardware Offload: Supported
HBA Port WWN: 2000001b2165a631
  Port Mode: Target
  Port ID: e00034
  VLAN ID: 7
  Link Name: net3
  MTU Size: 2500
  Primary MAC Address: 00:1b:21:65:a6:31
  FCoE MAC Address: 0e:fc:00:e0:00:34
  Promiscuous Mode: Off
  State: Online
  FIP Mode: On
  FCoE Hardware Offload: Supported
```

▼ 如何强制 FCoE 端口重新初始化

当 FCoE 端口需要重新初始化时，请使用下面的步骤。在向 FC SAN 添加新设备或由于 SAN 上存在故障设备时，您可能需要强制 FCoE 端口重新初始化。在许多情况下，此操作可解决 FC-SAN 中的问题。

在目标端口端发出此命令时，重置目标端口。从主机端口端发出此命令时，重置主机端口。

连接 FC 交换机时，SAN 中的其他 FC 端口会获得远程状态更改通知 (remote state change notification, RSCN)。此外，在此操作之后，其他启动器将始终重新发现该端口，并将建立或再次使用 FC 登录会话。此命令将打断 I/O，但 I/O 会继续。它不具备破坏性，因为它不会导致数据丢失。

1. 成为管理员。
2. 强制连接到端口的链接重新初始化。

例如：

```
# fcadm force-lip 200000144fc2d508
```

配置 FCoE 硬件负载转移

在具有 FCoE 端口的系统上，可以使用下表中列出的可调参数来降低 CPU 使用率及改善性能。Intel 10 Gb 以太网设备支持这些可调参数，这些参数在 `ixgbe.conf` 文件中设置。

将 `/kernel/drv/ixgbe.conf` 复制到 `/etc/driver/drv/ixgbe.conf`，然后针对您的 FCoE 环境修改可调参数值。

表 6-1 FCoE 硬件负载转移可调参数

可调参数	说明	范围	缺省值
<code>fcoe_txcrc_enable</code>	控制 <code>ixgbe</code> 驱动程序是否为传送的 FCoE 数据包转移 FC CRC 事务负载。	0 – 禁用 FC CRC 事务负载转移 1 – 启用 FC CRC 事务负载转移	1
<code>fcoe_lso_enable</code>	控制 <code>ixgbe</code> 驱动程序是否为传送的 FCoE 数据包转移 FC 大量传送事务负载。	0 – 禁用 FC LSO 事务负载转移 1 – 启用 FC LSO 事务负载转移	1
<code>fcoe_rxcrc_enable</code>	控制 <code>ixgbe</code> 驱动程序是否为接收的 FCoE 数据包转移 RC CRC 事务负载。	0 – 禁用 FC RX CRC 事务负载转移 1 – 启用 FC RX CRC 事务负载转移	1
<code>fcoe_lro_enable</code>	控制 <code>ixgbe</code> 驱动程序是否为接收的 FCoE 数据包转移 FC 大量接收事务负载。	0 – 禁用 FC LRO 事务负载转移 1 – 启用 FC LRO 事务负载转移	0

配置 SAS 域

本章介绍了 SAS 域的注意事项、SAS 设备搜索以及 SAS 引导设备配置。

本章包含以下主题：

- “SAS 多路径注意事项” [87]
- “动态搜索 SAS 设备” [87]
- “配置 SAS 引导设备” [88]

SAS 多路径注意事项

- 使用捆绑的 mpt 驱动程序时，在 Oracle Solaris 发行版中支持 SAS 多路径。
- SAS 扩展器在 Oracle Solaris 发行版中不受支持。
- 在 Oracle Solaris 发行版中不支持对 SAS 连接的设备使用 LUN 屏蔽。
- 当一台服务器尝试关闭一个设备而另一台服务器正尝试访问此设备时会出现意外结果，在连接到 SAS 域的系统上禁用电源管理可防止发生这种情况。有关电源管理的更多信息，请参见 [poweradm\(1M\)](#)。

动态搜索 SAS 设备

使用 mpt 驱动程序时可动态添加和删除 SAS 设备。您不再需要编辑 `sd.conf` 文件，系统即可检测特定目标与 mpt 驱动程序连接的 LUN。有关更多信息，请参见 [mpt\(7D\)](#) 和 [mpt_sas\(7D\)](#)。

如果您在 SAS 域中添加或删除设备，系统会向 `/var/adm/messages` 文件写入一条消息，表明添加或删除了设备。如果已添加了设备，可通过 `format` 命令识别此设备并使其可用。

配置 SAS 引导设备

运行 Oracle Solaris OS 的系统可从多路径 SAS 设备或连接到 SAS 控制器的 SATA 设备进行引导。仅捆绑的 mpt 驱动程序支持 Oracle Solaris 发行版本中的 SAS 多路径。

某些 SAS 和 SATA 设备可能不支持多路径，但仍可作为非多路径设备运行。有关更多信息，请访问 My Oracle Support 站点。

配置 IPFC SAN 设备

本章提供了主机系统上的基于光纤通道的 Internet 协议 (Internet Protocol over Fibre Channel, IPFC) 配置信息以介绍 SAN 中 IPFC 设备的识别和基于 FC 的 IP 的实现。IPFC 驱动程序基于 RFC 2625，允许 IP 通信在 FC 上运行。

本章包含以下主题：

- “IPFC 注意事项” [89]
- “调用和配置 IPFC” [92]

IPFC 注意事项

下表显示 IPFC 支持的功能。

表 8-1 IPFC (NFS/NAS 和 SNDR)

功能	受支持
级联	是，仅在使用光纤区域时
区域类型	光纤区域，HBA 配置为 F 端口点对点连接
每个区域的最大设备端口数	253

具有下列限制：

- Oracle 1 GB 交换机不支持 IPFC。
- 不支持混杂模式。不能使用 snoop 实用程序。
- 仅通过广播支持多播。
- 不能将使用 IPFC 的网卡用作路由器。缺省情况下，Oracle Solaris OS 中已禁用 IP 转发。
- 连接 IPFC 之后，可使用任意标准网络命令。在此环境下使用这些命令 (telnet、ping 或 ftp) 的方式与在以太网设置中相同。

确定光纤通道适配器端口实例

本节介绍如何为 IPFC 配置所需的主机系统。包括确定端口实例和检测 IPFC 实例这两个过程。

▼ 如何确定端口实例

1. 确定 HBA PCI 适配器插槽和 I/O 板 PCI 插槽。

需要插槽信息以计算光纤通道 (Fibre Channel, FC) 适配器端口实例。还可以使用以下命令确定卡所在的插槽以及与卡关联的编号：

```
prtdiag | grep -i pci
```

此过程中的示例假定您有这样一个阵列，其 HBA 卡位于 PCI 适配器插槽 5，PCI 适配器位于 I/O 板的插槽 1。

2. 确定实例编号。

- a. 在 `/etc/path_to_inst` 文件中搜索 `fp` 驱动程序绑定名称。

注 - 通过查找服务器硬件手册中所述的硬件路径确定正确的条目。

- b. 通过使用步骤 1 中的 I/O 板和插槽信息缩小搜索范围。

注 - 以下通过 HBA 在服务器中的物理位置来得出其设备路径的方法可能并不适用于 Oracle 的所有 Sun 服务器硬件。

- i. 用 PCI 适配器插槽号乘以适配器端口数。
例如，如果 HBA 有两个端口，则乘以 2。使用 HBA 位于 PCI 适配器插槽 5 的阵列，因此，5 乘以 2 得到 10。
- ii. 将 PCI 适配器 I/O 板插槽号加上步骤 i 得到的数值。
使用位于 PCI 适配器插槽 5 中的 HBA，I/O 板在 PCI 插槽 1，用 1 加上 10 得到总数 11。
- iii. 将步骤 ii 中得到的数值转换为十六进制。
数值 11 转换为十六进制中的 "b"。
- iv. 搜索带有 `pci@ hex` 的 `fp` 条目，其中 `hex` 是您在步骤 iii 中得到的数值。
下表显示具有以下路径的 PCI 单个 FC 网络适配器设备的设备路径的各个元素：

```
"/pci@b,2000/SUNW,qlc@2/fp@0,0" 7 "fp"
```

设备名	值
物理名称	/pci@b,2000/SUNW,qlc@2/fp@0,0
实例编号	7
驱动程序绑定名称	fp

- c. 手动创建各个 FP 实例。
在此示例中，*interface-number* 的值为 *fcip7*。

```
# ipadm create-ip fcip7
```

如果该命令成功，则控制台及消息文件中都会出现一条消息。例如：

```
Sep 13 15:52:30 bytownite ip: ip: joining multicasts failed (7) on fcip0 -  
will use link layer brocasts for multicast
```

▼ 如何创建 IPFC 实例

系统上的每个 FP 实例在 */dev/fc* 中都有一个条目。虽然 HBA 已经删除，但可能会存在一些过时链接。使用以下过程装入和创建 IPFC。

1. 针对 */dev/fc* 文件中的每个条目，显示通过该 HBA 端口可见的所有设备：

```
# luxadm -e dump_map /dev/fc/fp0  
Pos  Port_ID Hard_Addr Port WWN          Node WWN          Type  
0    610100  0           210000e08b049f53 200000e08b049f53 0x1f (Unknown Type)  
1    620d02  0           210000e08b02c32a 200000e08b02c32a 0x1f (Unknown Type)  
2    620f00  0           210000e08b03eb4b 200000e08b03eb4b 0x1f (Unknown Type)  
3    620e00  0           210100e08b220713 200100e08b220713 0x1f (Unknown Type,Host Bus Adapter)  
# luxadm -e dump_map /dev/fc/fp1  
No FC devices found. - /dev/fc/fp1
```

2. 根据设备列表，确定哪些目标 HBA 对于您想要与其建立 IPFC 通信的远程主机是可见的。

在此过程示例中，目标 HBA 的端口 ID 为 610100 和 620d02。原始 HBA 的端口 ID 为 620e00。

3. 列出可从中查看目标 HBA 端口的原始 HBA 端口的物理路径，其中 *originating-hba-link* 是步骤 2 中确定的链接的变量。

```
# ls -l /dev/fc/fp originating-hba-link
```

在下例中，0 是 *originating-hba-link* 的编号：

```
# ls -l /dev/fc/fp 0
```

```
lrwxrwxrwx 1 root root 51 Sep 4 08:23 /dev/fc/fp0 ->
../../devices/pci@8,600000/SUNW,qlc@1/fp@0,0:devctl
```

4. 搜索在步骤 3 中确定的物理路径。

您必须从路径名称输出中删除前导的 ../../devices。例如：

```
# grep pci@8,600000/SUNW,qlc@1/fp@0,0 /etc/path_to_inst
"/pci@8,600000/SUNW,qlc@1/fp@0,0" 0 "fp"
```

5. 从步骤 4 的命令输出确定原始 HBA 端口的 fp 实例。

实例编号在输出中 "fp" 的前面。在以下示例输出中，实例编号为 0。

```
"/pci@8,600000/SUNW,qlc@1/fp@0,0" 0 "fp"
```

6. 使用步骤 5 中的实例编号装入 IPFC 并创建 IPFC 接口。

在此示例中，实例为 0。

```
# ipadm create-ip fcip 0
```

调用和配置 IPFC

安装完毕，请立即使用 ipadm 命令手动启动 IPFC。您可对主机进行配置，以便在随后的重新引导中，IPFC 网络接口自动启动。本节介绍手动启动网络接口及将主机配置为在重新引导时自动检测的过程。

▼ 如何手动启动网络接口

当您想要检测具有特定网络掩码值的 IPFC 并让 IPFC 接口启动和运行时，请使用以下过程。

1. 成为管理员。

2. 配置适当的网络接口。

向您的网络管理员申请适当的 IP 地址和网络掩码信息。例如，要启用与 fp 实例 0 和 IP 地址 192.168.201.10 相关联的 IPFC 接口，请键入：

```
# ipadm create-ip fcip0
# ipadm create-addr -T static -a 192.168.201.10 fcip0/ipv4
```

有关更多信息，请参见 [ipadm\(1M\)](#)。

3. 确认网络可运行。

```
# ipadm show-if
```

▼ 如何将主机配置为自动网络配置

系统主机名在 `svc:/system/identity:node` 服务中设置。例如，主机名为 `sys-A`，IPFC 网络接口为 `fcip0`。

1. 成为管理员。
2. 确认已设置主机名。

```
# svccfg -s identity:node
svc:/system/identity:node> listprop config/nodename
config/nodename astring sys-A
```

如果需要设置主机名，请使用类似以下的语法：

```
# svccfg -s identity:node setprop config/nodename = "neo-1"
```

3. 将其他条目添加到 `/etc/inet/hosts` 文件。

安装程序将以最少条目创建 `/etc/inet/hosts` 文件。您必须使用文本编辑器手动添加其他条目。有关其他信息，请参见 [hosts\(4\)](#)。

`/etc/inet/hosts` 文件包含主机数据库。此文件包含主机名和主网络接口 IP 地址，以及附加到系统的其他网络接口和计算机必须了解的任何其他网络接口的 IP 地址。

以下示例所示为 `etc/inet/host` 文件。

```
127.0.0.1      localhost    localhost
192.168.200.70 neo-1      #This is the local host name
192.168.201.10 fcip0     #Interface to network 192.168.201.10
```

4. 确认已使用 `files nis` 为主机配置了名称服务 SMF 服务。

```
# svccfg
svc:> select network/nis/client:default
svc:/network/nis/client:default> select name-service/switch
svc:/system/name-service/switch> listprop config/host
config/host astring "files nis"
svc:/system/name-service/switch> quit
```


在基于 x86 的系统上从光纤通道设备引导 Oracle Solaris OS

本章介绍了如何在基于 x86 的系统上手动安装 Oracle Solaris OS，并使用 2 Gb 和 4 Gb 光纤通道 (Fibre Channel, FC) HBA 驱动程序实现 Solaris 多路径 I/O 功能。可在 Solaris 安装程序的磁盘选择部分中选择要从中执行引导的光纤通道 (Fibre Channel, FC) 设备。

本章包含以下主题：

- [“Oracle Solaris OS 安装要求” \[95\]](#)
- [“Oracle Solaris OS 安装概述” \[96\]](#)
- [“Oracle Solaris OS 安装过程” \[96\]](#)

Oracle Solaris OS 安装要求

要执行安装，您必须具备以下项：

- Oracle Solaris 11、Oracle Solaris 11.1 或 Oracle Solaris 11.2 各安装 DVD 中的任何一个 – 以下安装方法可用于在基于 x86 系统上配置 FC 设备：
 - 自动安装 – 在网络上安装多个客户机系统。可从用于安装单个系统的介质引导，但如果您要为多个客户机系统定制安装，需要一台安装服务器。
 - 文本安装 – 从介质或安装服务器安装单个系统。
- 连接到基于 x86 的系统的 FC HBA
- 用于基于网络的 OS 安装的 10/100/1000 兆字节/秒的以太网

Oracle 的 1 Gb HBA 不支持在基于 x86 的系统上通过 SAN 进行引导。Oracle 的 4 Gb HBA 支持在基于 x86 的系统上通过 SAN 进行引导。大多数 2 Gb HBA 都支持在基于 x86 的系统上通过 SAN 进行引导，但以下 HBA 除外：

- Oracle 的 StorageTek 2 Gb Enterprise Class 双端口光纤 HBA，SG-XPCI2FC-QF2-Z
 - Oracle 的 StorageTek 2 Gb PCI 双端口光纤 HBA，SG-XPCI2FC-QF2
 - Oracle 的 StorageTek 2 Gb FC PCI 单通道网络适配器，X6767A
 - Oracle 的 StorageTek 2 Gb FC PCI 双通道网络适配器，X6768A
 - Oracle 的 StorageTek 2 Gb FC PCI-X Enterprise 单端口 HBA，SG-XPCI1FC-QL2
 - Oracle 的 StorageTek 2 Gb FC PCI 单端口主机适配器，SG-XPCI1FC-QF2
-

Oracle Solaris OS 安装概述

为了成功地完成安装，请在安装过程的磁盘选择部分中使用基于 FC 的设备。在交互式安装结束时，您必须更改 x86 BIOS 和 FC HBA BIOS 以标识用于从远程多路径磁盘引导 Oracle Solaris 的 FC 启动器。

在安装 OS 之后和重新引导之前，可通过在新安装的逻辑单元号 (logical unit number, LUN) 上发出 `luxadm` 命令来收集配置信息。`luxadm` 命令输出会提供一个从 `c#t#d#` 到阵列 WWN 和 LUN 的映射。记录 HBA 全球名称 (World Wide Name, WWN) 和阵列 WWN 端口信息。

在重新引导期间，可使用 WWN 和 LUN 为每个 HBA 设置 HBA BIOS，系统将使用此 HBA BIOS 从同一阵列上的每个 LUN 进行引导。将系统 BIOS 从 CD-ROM 或网络引导更改为从磁盘引导。

注 - 需要访问控制台才能更改 HBA 和系统 BIOS。

另外，在安装 Oracle Solaris OS 期间，需要注意下列事项：

- 如果使用的是定制的磁盘布局，不要删除重叠 (s2) 分区。x86 引导安装程序与此分区相关联。
- 缺省情况下，Solaris I/O 多路径功能使用指向单个引导设备的多个路径来管理 FC 引导设备。

Oracle Solaris OS 安装过程

以下过程介绍了如何安装 Oracle Solaris OS。

▼ 如何安装 Oracle Solaris OS

1. 安装 HBA 硬件。
请遵循 <http://www.oracle.com/technetwork/documentation/oracle-storage-networking-190061.html> 上相应 Oracle HBA 安装指南中的说明。
2. 安装 Oracle Solaris OS。
如果选择自动安装，并且希望在安装期间选择要安装的具体设备，请参见《[安装 Oracle Solaris 11.2 系统](#)》。

▼ 如何执行 DVD 或基于网络的 OS 安装

在安装 HBA 后，请执行下列步骤在基于 x86 的系统上对 Oracle Solaris OS 进行 DVD 安装或基于网络的安装。

有关更多信息，请参阅《[安装 Oracle Solaris 11.2 系统](#)》。

1. 如果要从 DVD-ROM 而不是网络进行安装，请将 Oracle Solaris 软件 DVD 插入到 DVD-ROM 驱动器中。
2. 初始开机后，提供系统 BIOS 并将其配置为从网络或 DVD-ROM（根据具体情况）引导。
3. 通过选择下列一种方法安装 Oracle Solaris OS。
 - 自动安装 – 通过从基于 x86 的系统的 BIOS 中选择网络引导，即可开始自动安装。如果选择自动安装，请跳到第 7 步。
 - 文本安装 – 从介质引导或从安装服务器引导时通过从 GRUB 菜单中选择下列选项即可开始文本安装。

Oracle Solaris 11.2 Text Installer and command line

安装完成后，可退出安装程序并配置设备。

4. 文本安装 – 选择所需的阵列及其关联的 LUN。
5. 文本安装 – 通过从每个安装菜单中选择所需的安装选项继续安装。
6. 文本安装 – 在最后一个安装屏幕中，确认您的选择，然后开始安装 Oracle Solaris OS。
7. 在安装完成后，根据是执行自动安装还是文本安装选择下列一个选项。
 - 自动安装 – 缺省情况下，由于 `/usr/share/auto_install/default.xml` 文件中的下列清单关键字，在安装后系统不会重新引导。这表示您可以在系统重新引导之前配置您的设备。

```
<auto_install>
  <ai_instance name="default">
    .
    .
    .
```

如果前一次安装将下列关键字值设置为 true，请将此值更改为 false，以便在系统重新引导之前配置设备。

```
<auto_install>
  <ai_instance name="default" auto_reboot="true">
    .
    .
    .
```

- 文本安装 – 安装完成时，选择 "Quit"（退出）选项退出安装程序以配置设备。

8. 在安装完成之后重新引导之前，在安装期间选择的 LUN 上发出 `luxadm display` 命令。请参见图 9-1 “`luxadm display` 命令和输出”。

```
# luxadm display /dev/rdisk/c0t600015D0002028000000000000001142d0s2
DEVICE PROPERTIES for disk: /dev/rdisk/c0t600015D0002028000000000000001142d0s2
Vendor:      SUN
Product ID:  SE6920
Revision:    0202
Serial Num:  00500057
Unformatted capacity: 10240.000 MBytes
Read Cache:  Enabled
  Minimum prefetch:  0x0
  Maximum prefetch:  0xffff
Device Type:  Disk device
Path(s):

/dev/rdisk/c0t600015D0002028000000000000001142d0s2
/devices/scsi_vhci/disk@g600015d0002028000000000000001142:c,raw
Controller   /dev/cfg/c4
Device Address      213600015d207200,0
Host controller port WWN      210100e08b206812
Class             primary
State             ONLINE
Controller        /dev/cfg/c11
Device Address      213600015d207200,0
Host controller port WWN      210100e08b30a2f2
Class             primary
State             ONLINE
```

图 9-1 luxadm display 命令和输出



可以使用图中示例中的下列 luxadm 命令输出将基于 MPxIO 的 c#t#d# 映射至 HBA WWN 和阵列 WWN：

- MPxIO c#t#d# = c0t600015d000202800000000000000001142d0
- 阵列 WWN = 213600015d207200，LUN 0
- HBA WWN = 210100e08b206812 和 210100e08b30a2f2

9. 在重新引导过程中，注意查看监视器，以便进入 HBA #1 BIOS 屏幕，将引导设备指定为您刚刚在其中安装了 Oracle Solaris OS 的 FC LUN。
- 为每个要用于多路径的 HBA 执行此步骤，并将引导设备指定为您在其中安装了 Oracle Solaris OS 的 FC LUN。请参见图 9-2 “某个 HBA WWN 的 HBA BIOS 屏幕”和图 9-3 “从 FC LUN 启用引导的 HBA BIOS 屏幕”。

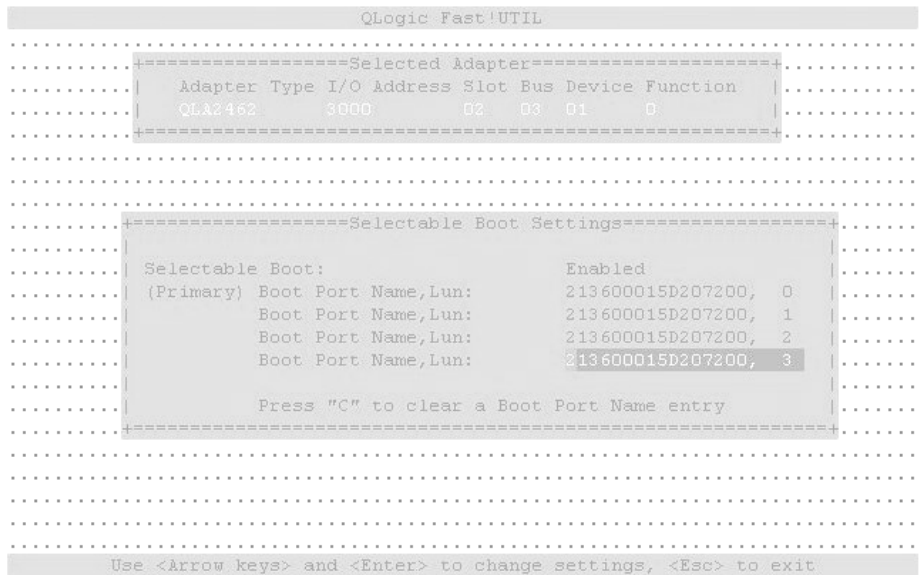
- 对于 QLogic HBA BIOS，请执行下列步骤。
 - a. 在主机重新引导期间，按 Ctrl-Q 组合键以显示 HBA BIOS 屏幕。
 - b. 选择要作为引导设备启用的 HBA 并启用引导。

- c. 配置引导设备：
 - i 选择配置设置。
 - ii 选择可选引导设置。
 - iii 确保 "Selectable Boot" (可选引导) 已设置为启用。
在此菜单中，可以按阵列 WWPN 选择引导设备/LUN。
 - iv 保存并退出 HBA BIOS 屏幕。
- 对于 Emulex HBA BIOS，请执行下列步骤。
 - a. 在主机重新引导期间，按 Alt-E 以显示 HBA BIOS 屏幕。
 - b. 选择要作为引导设备启用的 HBA 并启用引导。
 - c. 选择 "Configure Boot Devices" (配置引导设备)。
 - d. 选择引导项。
 - e. 选择所需引导设备的 WWPN。
 - f. 键入 LUN 编号。
 - g. 选择引导 LUN。
 - h. 选择带有阵列 WWPN 的引导设备。
 - i. 保存并退出 HBA BIOS 屏幕。

图 9-2 某个 HBA WWN 的 HBA BIOS 屏幕



图 9-3 从 FC LUN 启用引导的 HBA BIOS 屏幕



注 - 上图显示了下列修改：

- Selectable boot = Enabled
- ARRAY WWN = 213600015d207200
- ARRAY LUN = 0
- HBA WWN = 210100e08b206812

10. 为所有 HBA 和所有 LUN 重复相应的修改。
11. 键入与供应商访问方法对应的系统 BIOS，并将引导设备指定为您在其中安装了 Oracle Solaris OS 的 FC LUN。
12. 通过使用系统 BIOS 中指定的 FC LUN 重新引导至新安装的 Oracle Solaris OS。

磁带设备的持久绑定

本章介绍如何为磁带设备创建持久绑定以确保 SAN 上的多台服务器中的 `/dev` 条目是相同的。

本章包含以下主题：

- [“持久绑定概述” \[103\]](#)
- [“创建磁带链接” \[104\]](#)

持久绑定概述

为了简化基于 SAN 的数据中心中的服务器管理，Oracle 在 Solaris OS 中提供的 StorageTek™ SAN Foundation 软件栈会动态检测 SAN 中的设备，并建立关联的 `/dev` 树条目，而无需您编辑配置文件。

在大多数情况下，此过程大大简化了 SAN 管理。但是，对于磁带设备，您可能希望能够显式指定 `/dev` 条目的创建方式，并且能够确保 `/dev` 条目在 SAN 上的多台服务器中是相同的。本章介绍了如何在 Solaris OS 中指定此磁带绑定，同时又保留自动搜索基于磁盘的设备的优势。

`/dev/rmt` 目录包含指向磁带设备在 `/devices` 下的物理设备的链接。系统以 `/dev/rmt/N`、`/dev/rmt/Nb` 和 `/dev/rmt/Nbn` 形式（其中 `N` 为从 0 开始的整数计数器）用 24 个次要节点表示它所发现的每个磁带 LUN。此数字在新设备枚举期间由 `devfsadm` 拾取。`devfsadm` 发现的每个新磁带逻辑单元号 (Logical Unit Number, LUN) 在 `/dev/rmt` 中都会获得下一个可用的编号。

由于 `/dev/rmt` 名称依赖于设备在设备树中的显示顺序，所以在不同系统上会有所不同。对于由两个或更多不同系统发现的特定磁带设备，在其中每个系统上 `/dev/rmt` 链接可能都不相同。对于 Symantec (VERITAS) NetBackup (SSO 选项) 的最常见用法，这种不同可能会引起问题。而且，如果更换此驱动器，链接也会发生更改，除非供应商提供了方法来保留驱动器的端口全球名称 (Port World-Wide Name, *PWWN*)。

创建磁带链接

/etc/devlink.tab 文件被称为缺省的设备表文件。它指定了 devfsadm 用来在 /dev 目录中创建链接的规则。此文件不包含任何磁带的条目，因为 devfsadm 已经能够为磁带驱动器创建链接，但可以添加规则，以修改创建磁带链接的缺省行为。有关更多信息，请参见 [devlinks\(1M\)](#)。

对于 Oracle Solaris OS 可发现但 devlink 文件中未指定的任何磁带驱动器，devfsadm 会自动指定一个从 0 开始的次要节点号。这些次要节点号可能与 /etc/devlink.tab 中手动指定的任何较低编号相冲突，所以请务必指定足够大的编号以避免发生冲突。

这种方法容易导致 /dev/rmt 中出现重复的链接。在 /etc/devlink.tab 中指定条目之前搜索到的任何磁带都已自动创建了链接。在添加了条目并运行 devfsadm 时，原始链接继续驻留在 /dev/rmt 中，导致出现重复的链接。要删除 /dev/rmt 中的原始链接，请先运行 `rm /dev/rmt/*` 命令，之后再运行 devfsadm。

这种方法不能用于连接到多个 HBA 端口的多端口磁带驱动器。如果多个 HBA 端口都连接到相同的磁带 LUN，系统会检测到两个而不是一个磁带驱动器。prtconf 输出中最后显示的磁带驱动器会获得 /etc/devlink.tab 生成的链接。

以下示例显示了 devlink.tab 文件中一个样例磁带条目。

```
type=ddi_byte:tape;addr=PWWN,LUN-number; rmt/rmt-number\M0
```

按照 /dev/rmt/ *N* 更改 rmt #。然后更改 PWWN 和 LUN 以与所需磁带设备相匹配。如下所示，通过在现有的 /dev/rmt/ 链接上运行 `ls -l` 命令，您可以获得此值。

```
# ls -l /dev/rmt/4
lrwxrwxrwx 1 root root 69 Oct 6 14:57 /dev/rmt/4 ->
../../../../devices/pci@1f,700000/SUNW,qlc@2/fp@0,0/st@w5005076300617717,0:
```

例如，如果您希望 /dev/rmt/ 编号为 40，可在 /etc/devlink.tab 中创建一个如下例所示的条目：

```
# type=ddi_byte:tape;addr=w5005076300617717,0; rmt/40\M0
```

然后可以将此行添加到 SAN 上每台将使用此驱动器的 Solaris 服务器的 devlink 文件中，这样它便能始终显示为次要编号 40。

▼ 如何创建磁带设备链接

1. 成为管理员。
2. 按照“[创建磁带链接](#)” [104] 中的描述在 /etc/devlink.tab 中创建条目。

如果 `devfsadm` 以前已搜索到了设备，您必须通过在现有链接上运行 `ls -l` 命令来确定设备地址。

注 - 如上所述，请务必指定 `/dev/rmt/ N` 编号以避免与任何自动配置的设备发生冲突。

3. 通过运行 `rm /dev/rmt/*` 命令从 `/dev/rmt` 删除现有的链接。

4. 运行 `devfsadm`。

此命令除了自动为任何未指定的设备创建链接外，还会根据 `/etc/devlink.tab` 中的条目创建新链接。

光纤连接的设备的手动配置

本附录说明如何在 Oracle Solaris OS 中配置和取消配置光纤设备，阐述了如何在启用和未启用多路径软件的情况下检测并配置主机上的可见光纤设备。

本章包含以下主题：

- “手动配置光纤连接的设备” [107]
- “配置光纤设备节点” [108]
- “在未启用多路径的情况下配置设备节点” [110]
- “在启用 Solaris 多路径的情况下配置设备节点” [112]
- “取消配置光纤设备” [116]

手动配置光纤连接的设备

在 Oracle Solaris 发行版中，光纤连接的设备自动供 Oracle Solaris 系统使用。

如果您希望手动配置光纤连接的设备，请使用以下步骤来更改缺省行为。

注 - 更改缺省行为会使您的所有光纤连接的设备不可用，从而导致必须在引导时可用的光纤连接的设备发生问题。

▼ 如何手动配置光纤连接的设备

1. 成为管理员。
2. 将 `/kernel/drv/fp.conf` 文件复制到 `/etc/driver/drv/fp.conf` 文件。
3. 确保取消注释 `/etc/driver/drv/fp.conf` 文件中的以下行，以启用手动配置。

```
manual_configuration_only=1;
```

有关该设置的更多信息，请参见 [cfgadm_fp\(1M\)](#) 和 [fp\(7d\)](#)。

4. 重新引导系统。
5. 对于要使之可用的每个光纤连接的设备，选择以下任务之一，具体选择哪个任务取决于您是否使用 Solaris I/O 多路径功能。
 - “在未启用多路径的情况下配置设备节点” [110]
 - “在启用 Solaris 多路径的情况下配置设备节点” [112]

如果需要光纤连接的设备的原始缺省行为，请参见下一步。

6. 确保注释 `/etc/driver/drv/fp.conf` 文件中的以下行，以禁用手动配置：

```
# manual_configuration_only=1;
```

7. 重新引导系统。

```
# init 6
```

配置光纤设备节点

在直接连接的系统或 SAN 中配置硬件后，必须确保系统识别这些设备。本节介绍主机对光纤设备（在 SAN 中也称为 24 位 FC 寻址设备）的识别。在 SAN 中配置设备、端口和区域后，确保系统识别这些设备。在具有 FC 支持的 SAN 中，您最多可以将 1600 万台光纤设备连接在一起。

本节仅限于从 Oracle Solaris OS 角度需要的操作。不涵盖其他方面，例如，设备可用性和特定于设备的管理。如果设备由其他软件（如卷管理器）管理，请参阅卷管理器产品文档，了解更多说明。

确保 LUN 级别信息可见

▼ 如何确保 LUN 级别信息可见

1. 成为管理员。
2. 确定 LUN 级别信息。

```
# cfgadm -al -o show_SCSI_LUN
```

如果您在系统引导之后立即发出 `cfgadm -al -o show_SCSI_LUN controller-ID` 命令，则输出可能不显示光纤通道协议 (Fibre Channel Protocol, FCP) SCSI LUN 级别信息。之所

以不显示此信息是因为存储设备驱动程序（例如，`ssd` 和 `st` 驱动程序）还没有装入正在运行的系统上。

3. 确定是否装入了这些驱动程序。

例如：

```
# modinfo | grep ssd
```

装入驱动程序后，LUN 级别信息便会显示在 `cfgadm` 输出中。

▼ 如何检测系统上的可见光纤设备

本节提供了使用 FC 主机端口 `c0` 和 `c1` 检测光纤设备的过程示例。此过程还给出了使用 `cfgadm` 命令显示的设备配置信息。

注 - 在以下示例中，仅列出了故障转移路径连接点 ID (`Ap_Id`)。系统上显示的 `Ap_Id` 取决于系统配置。

1. 成为管理员。
2. 显示有关系统上的连接点的信息。

```
# cfgadm -l
```

Ap_Id	Type	Receptacle	Occupant	Condition
c0	fc-fabric	connected	unconfigured	unknown
c1	fc-private	connected	configured	unknown

在本例中，`c0` 表示一个光纤连接的主机端口，`c1` 表示一个专用的循环连接的主机端口。使用 `cfgadm` 命令管理光纤连接的主机端口上的设备配置。

缺省情况下，专用的循环连接的主机端口上的设备配置由运行 Oracle Solaris OS 的系统管理。

3. 显示有关主机端口及其连接设备的信息。

```
# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Condition
c0	fc-fabric	connected	unconfigured	unknown
c0::50020f2300006077	disk	connected	unconfigured	unknown
c0::50020f23000063a9	disk	connected	unconfigured	unknown
c0::50020f2300005f24	disk	connected	unconfigured	unknown
c0::50020f2300006107	disk	connected	unconfigured	unknown
c1	fc-private	connected	configured	unknown
c1::220203708b69c32b	disk	connected	configured	unknown
c1::220203708ba7d832	disk	connected	configured	unknown
c1::220203708b8d45f2	disk	connected	configured	unknown
c1::220203708b9b20b2	disk	connected	configured	unknown

注 - `cfgadm -l` 命令显示有关 FC 主机端口的信息。您还可以使用 `cfgadm -al` 命令来显示 FC 设备的相关信息。与 `c0` 相关联的 `Ap_Id` 字段中包含端口全球名称 (world wide name, WWN) 的行表示光纤设备。使用 `cfgadm configure` 和 `unconfigure` 命令管理这些设备，并使这些设备对使用 Oracle Solaris OS 的系统可用。`c1` 下具有端口 WWN 的 `Ap_Id` 设备表示通过 `c1` 主机端口配置的专用循环设备。

在未启用多路径的情况下配置设备节点

本节介绍没有启用多路径的系统上的光纤设备配置任务。

本节中的过程显示如何检测系统上的可见光纤设备，以及如何配置这些设备使其对运行 Oracle Solaris OS 的系统可用。本节中的过程以特定设备为例说明如何使用 `cfgadm` 命令来检测和配置光纤设备。

由您提供并使用 `cfgadm` 命令显示的设备信息取决于系统配置。

▼ 如何在不使用多路径的情况下手动配置 FC 设备

以下样例过程介绍如何配置连接到光纤连接的主机端口 `c0` 的光纤设备。

1. 成为管理员。
2. 确定要配置的设备。

```
# cfgadm -al
Ap_Id          Type      Receptacle  Occupant    Condition
c0              fc-fabric connected   unconfigured unknown
c0::50020f2300006077 disk      connected   unconfigured unknown
c0::50020f23000063a9 disk      connected   unconfigured unknown
c0::50020f2300005f24 disk      connected   unconfigured unknown
c0::50020f2300006107 disk      connected   unconfigured unknown
c1              fc-private connected   configured   unknown
c1::220203708b69c32b disk      connected   configured   unknown
c1::220203708ba7d832 disk      connected   configured   unknown
c1::220203708b8d45f2 disk      connected   configured   unknown
c1::220203708b9b20b2 disk      connected   configured   unknown
```

3. 配置光纤设备。

```
# cfgadm -c configure c0::50020f2300006077
```

4. 检验是否配置了选定光纤设备。

```
# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Condition
c0	fc-fabric	connected	configured	unknown
c0::50020f2300006077	disk	connected	configured	unknown
c0::50020f23000063a9	disk	connected	unconfigured	unknown
c0::50020f2300005f24	disk	connected	unconfigured	unknown
c0::50020f2300006107	disk	connected	unconfigured	unknown
c1	fc-private	connected	configured	unknown
c1::220203708b69c32b	disk	connected	configured	unknown
c1::220203708ba7d832	disk	connected	configured	unknown
c1::220203708b8d45f2	disk	connected	configured	unknown
c1::220203708b9b20b2	disk	connected	configured	unknown

请注意，c0 和 c0::50020f2300006077 的 Occupant 列均显示为 configured，表明 c0 端口配置了一个插卡并配置了 c0::50020f2300006077 设备。

5. 显示多 LUN SCSI 设备的 FCP SCSI LUN 信息。

以下代码示例显示通过 Ap_Id c0:50020f2300006077 连接的物理设备配置了四个 LUN。

```
# cfgadm -al -o show_SCSI_LUN c0
```

Ap_Id	Type	Receptacle	Occupant	Condition
c0	fc-fabric	connected	configured	unknown
c0::50020f2300006077,0	disk	connected	configured	unknown
c0::50020f2300006077,1	disk	connected	configured	unknown
c0::50020f2300006077,2	disk	connected	configured	unknown
c0::50020f2300006077,3	disk	connected	configured	unknown

此设备目前在运行 Oracle Solaris OS 的系统上可用。路径表示 c0::50020f2300006077 所代表的物理设备中的每个 SCSI LUN。

▼ 如何在没有使用多路径的情况下配置多个 FC 设备

确保先使用“确保 LUN 级别信息可见”[108]过程确定系统中的可见设备。以下过程介绍如何配置所有连接到光纤连接的主机端口的未配置光纤设备。用作示例的端口为 c0。

1. 成为管理员。
2. 确定要配置的设备。

```
# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Condition
c0	fc-fabric	connected	unconfigured	unknown
c0::50020f2300006077	disk	connected	unconfigured	unknown
c0::50020f23000063a9	disk	connected	unconfigured	unknown
c0::50020f2300005f24	disk	connected	unconfigured	unknown
c0::50020f2300006107	disk	connected	unconfigured	unknown
c1	fc-private	connected	configured	unknown
c1::220203708b69c32b	disk	connected	configured	unknown
c1::220203708ba7d832	disk	connected	configured	unknown
c1::220203708b8d45f2	disk	connected	configured	unknown

```
c1::220203708b9b20b2 disk connected configured unknown
```

3. 配置选定端口上的所有未配置设备。

```
# cfgadm -c configure c0
```

注 - 此操作将对 c0 上的所有设备重复单个设备的 configure 操作。如果 c0 上的设备数量巨大，此操作可能很耗时。

4. 检验是否配置了 c0 上的所有设备。

```
# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Condition
c0	fc-fabric	connected	configured	unknown
c0::50020f2300006077	disk	connected	configured	unknown
c0::50020f23000063a9	disk	connected	configured	unknown
c0::50020f2300005f24	disk	connected	configured	unknown
c0::50020f2300006107	disk	connected	configured	unknown
c1	fc-private	connected	configured	unknown
c1::220203708b69c32b	disk	connected	configured	unknown
c1::220203708ba7d832	disk	connected	configured	unknown
c1::220203708b8d45f2	disk	connected	configured	unknown
c1::220203708b9b20b2	disk	connected	configured	unknown

5. 显示多 LUN SCSI 设备的 FCP SCSI LUN 信息。

以下代码示例显示 c0::50020f2300006077 和 c0::50020f2300006107 所表示的物理设备分别配置了四个 LUN。c0::50020f23000063a9 和 c0::50020f2300005f24 所表示的物理设备分别配置了两个 LUN。

```
# cfgadm -al -o show_SCSI_LUN c0
```

Ap_Id	Type	Receptacle	Occupant	Condition
c0	fc-fabric	connected	configured	unknown
c0::50020f2300006077,0	disk	connected	configured	unknown
c0::50020f2300006077,1	disk	connected	configured	unknown
c0::50020f2300006077,2	disk	connected	configured	unknown
c0::50020f2300006077,3	disk	connected	configured	unknown
c0::50020f23000063a9,0	disk	connected	configured	unknown
c0::50020f23000063a9,1	disk	connected	configured	unknown
c0::50020f2300005f24,0	disk	connected	configured	unknown
c0::50020f2300005f24,1	disk	connected	configured	unknown
c0::50020f2300006107,0	disk	connected	configured	unknown
c0::50020f2300006107,1	disk	connected	configured	unknown
c0::50020f2300006107,2	disk	connected	configured	unknown
c0::50020f2300006107,3	disk	connected	configured	unknown

在启用 Solaris 多路径的情况下配置设备节点

本节介绍如何对已启用多路径功能的系统执行光纤设备配置步骤。

缺省情况下，不配置连接到光纤连接的主机端口的设备，因此系统无法使用这些设备。使用 `cfgadm configure` 和 `cfgadm unconfigure` 命令来管理光纤设备的设备节点创建。有关更多信息，请参见 [cfgadm_fp\(1M\)](#)。本节中的过程说明如何检测系统上的可见光纤设备，以及如何将这些设备配置为多路径设备使其对系统可用。

由您提供并使用 `cfgadm` 命令显示的设备信息取决于系统配置。

▼ 如何配置单个多路径 FC 设备

以下样例过程使用光纤连接的主机端口 `c0` 和 `c2` 来在已启用多路径软件的系统上将光纤设备配置为多路径设备。

不论是否启用了多路径，`cfgadm -c configure` 命令对于光纤设备的效果都是相同的。

1. 成为管理员。
2. 确定要配置为多路径设备的设备的端口 WWN。

在光纤连接的主机端口上查找标记为 `fc-fabric` 的设备。您可以使用 `cfgadm -c configure` 命令来配置这些设备。

```
# cfgadm -al
Ap_Id          Type      Receptacle Occupant  Condition
c0             fc-fabric connected unconfigured unknown
c0::50020f2300006077 disk      connected unconfigured unknown
c0::50020f23000063a9 disk      connected unconfigured unknown
c1             fc-private connected configured unknown
c1::220203708b69c32b disk      connected configured unknown
c1::220203708ba7d832 disk      connected configured unknown
c1::220203708b8d45f2 disk      connected configured unknown
c1::220203708b9b20b2 disk      connected configured unknown
c2             fc-fabric connected unconfigured unknown
c2::50020f2300005f24 disk      connected unconfigured unknown
c2::50020f2300006107 disk      connected unconfigured unknown
```

在上例中，`c0::50020f2300006077` 和 `c2::50020f2300006107` `Ap_Id` 表示存储设备控制器的端口 WWN 不同的同一存储设备。`c0` 和 `c2` 主机端口启用了多路径。

3. 配置光纤设备并使这些设备对系统可用。

```
# cfgadm -c configure c0::50020f2300006077 c2::50020f2300006107
```

4. 检验是否配置了选定的设备。

```
# cfgadm -al
Ap_Id          Type      Receptacle Occupant  Condition
c0             fc-fabric connected configured unknown
```

```

c0::50020f2300006077 disk      connected  configured  unknown
c0::50020f23000063a9 disk      connected  unconfigured unknown
c1                fc-private connected  configured  unknown
c1::220203708b69c32b disk      connected  configured  unknown
c1::220203708ba7d832 disk      connected  configured  unknown
c1::220203708b8d45f2 disk      connected  configured  unknown
c1::220203708b9b20b2 disk      connected  configured  unknown
c2                fc-fabric  connected  configured  unknown
c2::50020f2300005f24 disk      connected  unconfigured unknown
c2::50020f2300006107 disk      connected  configured  unknown

```

请注意，c0 和 c0::50020f2300006077 的 Occupant 列指定 configured，这表明 c0 端口至少配置了一个插卡并配置了 c0::50020f2300006077 设备。在 c2 和 c2::50020f2300006107 中已进行了相同的更改。

如果成功完成了配置操作，系统上即创建了多路径设备。如果 c0::50020f2300006077 和 c2::50020f2300006107 表示的物理设备配置了多个 SCSI LUN，则每个 LUN 都配置为一个多路径设备。下例显示通过 c0::50020f2300006077 和 c2::50020f2300006107 配置了两个 LUN。每个 Ap_Id 与多路径设备的一个路径相关联。

```

# cfgadm -al -o show_SCSI_LUN c0::50020f2300006077\ c2::50020f2300006107
Ap_Id      Type      Receptacle  Occupant  Condition
c0::50020f2300006077,0 disk      connected  configured unknown
c0::50020f2300006077,1 disk      connected  configured unknown
c2::50020f2300006107,0 disk      connected  configured unknown
c2::50020f2300006107,1 disk      connected  configured unknown

```

上例表示为 c0::50020f2300006077 和 c2::50020f2300006107 表示的设备创建了以下两个多路径设备。

▼ 如何配置多个多路径 FC 设备

在配置或删除设备节点之前，请务必先使用“[确保 LUN 级别信息可见](#)” [108] 过程确定光纤设备。

在本例中，光纤连接的主机端口上的 Ap_Id 是多路径设备的路径。例如，将配置路径通过 c2 的所有设备，而不配置任何路径通过 c0 的设备。c2 是从系统到光纤的连接点，而 c2::50020f2300006107 是从存储到光纤的连接点。系统会检测其配置所针对的光纤中的所有存储设备。

在已经通过一个 Ap_Id 配置的设备上配置另一个 Ap_Id 会导致先前配置的设备出现一条附加路径。此时，不会创建新设备节点。只有首次配置相应设备的 Ap_Id 时才会创建设备节点。

1. 成为管理员。
2. 确定要配置的光纤连接的主机端口。

```
# cfgadm -al
Ap_Id              Type      Receptacle  Occupant  Condition
c0                  fc-fabric connected  configured unknown
c0::50020f2300006077 disk      connected  configured unknown
c0::50020f23000063a9 disk      connected  configured unknown
c1                  fc-private connected  configured unknown
c1::220203708b69c32b disk      connected  configured unknown
c1::220203708ba7d832 disk      connected  configured unknown
c1::220203708b8d45f2 disk      connected  configured unknown
c1::220203708b9b20b2 disk      connected  configured unknown
c2                  fc-fabric connected  unconfigured unknown
c2::50020f2300005f24 disk      connected  unconfigured unknown
c2::50020f2300006107 disk      connected  unconfigured unknown
```

由 Ap_Id c0::50020f2300006077 和 c2::50020f2300006107 表示的设备是已经配置了 c0::50020f2300006077 的同一物理设备的两个路径。配置选定端口上的未配置设备。此操作将对 c2 上的所有设备重复单个设备的 configure 命令。如果 c2 上的设备数量巨大，此操作可能很耗时。

```
# cfgadm -c configure c2
```

3. 检验是否配置了 c2 上的所有设备。

```
# cfgadm -al
Ap_Id              Type      Receptacle  Occupant  Condition
c0                  fc-fabric connected  configured unknown
c0::50020f2300006077 disk      connected  configured unknown
c0::50020f23000063a9 disk      connected  configured unknown
c1                  fc-private connected  configured unknown
c1::220203708b69c32b disk      connected  configured unknown
c1::220203708ba7d832 disk      connected  configured unknown
c1::220203708b8d45f2 disk      connected  configured unknown
c1::220203708b9b20b2 disk      connected  configured unknown
c2                  fc-fabric connected  configured unknown
c2::50020f2300005f24 disk      connected  configured unknown
c2::50020f2300006107 disk      connected  configured unknown
```

请注意，c2 的 Occupant 列和 c2 下的所有设备均被标记为 configured。

show_SCSI_LUN 命令显示多个 LUN SCSI 设备的 FCP SCSI LUN 信息。以下代码示例显示通过 c2::50020f2300006107 和 c2::50020f2300005f24 连接的物理设备分别配置了两个 LUN。

```
# cfgadm -al -o show_SCSI_LUN c2
Ap_Id              Type      Receptacle  Occupant  Condition
c2                  fc-fabric connected  configured unknown
c2::50020f2300005f24,0 disk      connected  configured unknown
c2::50020f2300005f24,1 disk      connected  configured unknown
c2::50020f2300006107,0 disk      connected  configured unknown
c2::50020f2300006107,1 disk      connected  configured unknown
```

取消配置光纤设备

本节提供有关取消配置光纤设备的多路径功能的信息。

取消对光纤设备的配置

在取消对光纤设备的配置之前，请停止设备的所有活动并卸载光纤设备上的所有文件系统。有关卸载说明，请参见 Oracle Solaris 管理文档。如果设备受任何卷管理器的控制，请在取消对设备的配置之前先参见卷管理器文档。

▼ 如何手动取消对 FC 设备的配置

以下过程介绍如何取消配置连接到光纤连接的主机端口 c0 的光纤设备。

1. 成为管理员。
2. 确定要取消配置的设备。
只能取消配置光纤连接的主机端口上的设备。

```
# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Condition
c0	fc-fabric	connected	configured	unknown
c0::50020f2300006077	disk	connected	configured	unknown
c0::50020f23000063a9	disk	connected	configured	unknown
c1	fc-private	connected	configured	unknown
c1::220203708b69c32b	disk	connected	configured	unknown
c1::220203708ba7d832	disk	connected	configured	unknown

3. 取消对光纤设备的配置。

```
# cfgadm -c unconfigure c0::50020f2300006077
```

4. 检验是否取消了对选定光纤设备的配置。

```
# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Condition
c0	fc-fabric	connected	configured	unknown
c0::50020f2300006077	disk	connected	unconfigured	unknown
c0::50020f23000063a9	disk	connected	configured	unknown
c1	fc-private	connected	configured	unknown
c1::220203708b69c32b	disk	connected	configured	unknown
c1::220203708ba7d832	disk	connected	configured	unknown

▼ 如何取消配置 FC 主机端口上的所有 FC 设备

以下过程介绍如何取消配置连接到光纤连接的主机端口的所有已配置光纤设备。

1. 成为管理员。
2. 确定要取消配置的光纤设备。
只能取消配置光纤连接的主机端口上的设备。

```
# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Condition
c0	fc-fabric	connected	configured	unknown
c0::50020f2300006077	disk	connected	configured	unknown
c0::50020f23000063a9	disk	connected	configured	unknown
c1	fc-private	connected	configured	unknown
c1::220203708b69c32b	disk	connected	configured	unknown
c1::220203708ba7d832	disk	connected	configured	unknown

3. 停止选定端口上的各光纤设备的所有活动，并卸载每个光纤设备上的所有文件系统。
如果设备受任何卷管理器的控制，请在取消对设备的配置之前先参见卷管理器文档。

```
# cfgadm -c unconfigure c0
```

4. 取消配置选定端口上的所有已配置光纤设备。

注 - 此操作将对 c0 上的所有设备重复单个设备的 unconfigure 操作。如果 c0 上的设备数量巨大，此过程可能很耗时。

5. 检验是否取消了对 c0 上的所有设备的配置。

```
# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Condition
c0	fc-fabric	connected	unconfigured	unknown
c0::50020f2300006077	disk	connected	unconfigured	unknown
c0::50020f23000063a9	disk	connected	unconfigured	unknown
c1	fc-private	connected	configured	unknown
c1::220203708b69c32b	disk	connected	configured	unknown
c1::220203708ba7d832	disk	connected	configured	unknown

请注意，c0 的 Occupant 列和与其连接的所有光纤设备显示为 unconfigured。

▼ 如何取消对多路径 FC 设备的配置

以下过程显示光纤连接的主机端口 c0 和 c2，说明如何取消配置与多路径设备关联的光纤设备。

1. 成为管理员。
2. 确定要取消配置的光纤设备的端口 WWN。

```
# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Condition
c0	fc-fabric	connected	configured	unknown
c0::50020f2300006077	disk	connected	configured	unknown
c0::50020f23000063a9	disk	connected	configured	unknown
c1	fc-private	connected	configured	unknown
c1::220203708b69c32b	disk	connected	configured	unknown
c1::220203708ba7d832	disk	connected	configured	unknown
c2	fc-fabric	connected	configured	unknown
c2::50020f2300005f24	disk	connected	configured	unknown
c2::50020f2300006107	disk	connected	configured	unknown

在本例中，c0::50020f2300006077 和 c2::50020f2300006107 Ap_Id 表示与某一多路径设备相关联的同一设备的不同端口 WWN。启用了 c0 和 c2 主机端口。

3. 停止选定端口上的各光纤设备的所有设备活动，并卸载各光纤设备上的所有文件系统。如果设备受任何卷管理器的控制，请参见卷管理器文档以维护光纤设备。
4. 取消配置与设备相关联的光纤设备。

通过 `cfgadm -c unconfigure` 命令只能取消配置光纤连接的主机端口上的设备。

```
# cfgadm -c unconfigure c0::50020f2300006077 c2::50020f2300006107
```

注 - 您最多可以分别从八个路径删除一台设备，例如，命令 `cfgadm -c unconfigure c0::1111, c1::2222, c3::3333`，依此类推。也可以从主机删除整组路径，如 `cfgadm -c unconfigure c0` 示例所示。

5. 检验是否取消了对选定设备的配置。

```
# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Condition
c0	fc-fabric	connected	configured	unknown
c0::50020f2300006077	disk	connected	unconfigured	unknown
c0::50020f23000063a9	disk	connected	configured	unknown
c1	fc-private	connected	configured	unknown
c1::220203708b69c32b	disk	connected	configured	unknown
c1::220203708ba7d832	disk	connected	configured	unknown
c2	fc-fabric	connected	configured	unknown
c2::50020f2300005f24	disk	connected	configured	unknown
c2::50020f2300006107	disk	connected	unconfigured	unknown

请注意，Ap_Id c0::50020f2300006077 和 c2::50020f2300006107 的配置已被取消。c0 和 c2 的 Occupant 列仍将这些端口显示为 configured，因为它们已配置了其他插卡。

系统将无法再使用与 Ap_Id c0::50020f2300006077 和 c2::50020f2300006107 关联的多路径设备。以下两个设备已从系统中删除：

```
/dev/rdisk/c6t60020F20000061073AC8B52D000B74A3d0s2
```

```
/dev/rdisk/c6t60020F20000061073AC8B4C50004ED3Ad0s2
```

▼ 如何取消配置多路径 FC 设备的一个路径

与前一节中的过程相反，本过程显示如何取消配置与 `c2::50020f2300006107` 相关联的一个设备，同时保留配置另一个设备 (`50020f2300006077`)。通过 `cfgadm unconfigure` 命令只能取消配置光纤连接的主机端口上的设备。

1. 成为管理员。
2. 确定要取消配置的多路径设备的 `Ap_Id`。

```
# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Condition
c0	fc-fabric	connected	configured	unknown
c0::50020f2300006077	disk	connected	configured	unknown
c0::50020f23000063a9	disk	connected	configured	unknown
c1	fc-private	connected	configured	unknown
c1::220203708b69c32b	disk	connected	configured	unknown
c1::220203708ba7d832	disk	connected	configured	unknown
c2	fc-fabric	connected	configured	unknown
c2::50020f2300005f24	disk	connected	configured	unknown
c2::50020f2300006107	disk	connected	configured	unknown

在本例中，`c0::50020f2300006077` 和 `c2::50020f2300006107` `Ap_Id` 表示同一设备的不同端口 WWN。

3. 取消配置与设备相关联的 `Ap_Id`。

注 - 如果 `Ap_Id` 表示设备的最后一个已配置路径，请停止该路径的所有活动，并卸载其上的所有文件系统。如果多路径设备受任何卷管理器的控制，请参见卷管理器文档以维护光纤设备。

在随后的示例中，表示为 `c2::50020f2300006107` 的路径被取消配置，而 `c0::50020f2300006077` 仍处于配置状态，显示了如何仅取消配置某一多路径设备的其中一个路径。

```
# cfgadm -c unconfigure c2::50020f2300006107
```

4. 检验选定路径 `c2::50020f2300006107` 是否已取消配置。

```
# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Condition
c0	fc-fabric	connected	configured	unknown
c0::50020f2300006077	disk	connected	configured	unknown
c0::50020f23000063a9	disk	connected	configured	unknown
c1	fc-private	connected	configured	unknown
c1::220203708b69c32b	disk	connected	configured	unknown
c1::220203708ba7d832	disk	connected	configured	unknown
c2	fc-fabric	connected	configured	unknown
c2::50020f2300005f24	disk	connected	configured	unknown

```
c2::50020f2300006107 disk          connected    unconfigured unknown
```

系统仍可通过另一个路径（表示为 `c0::50020f2300006077`）使用与该 Ap_Id 相关联的设备。一台设备可以连接到多个 Ap_Id，一个 Ap_Id 可以连接到多台设备。

```
/dev/rdisk/c6t60020F20000061073AC8B52D000B74A3d0s2
and
/dev/rdisk/c6t60020F20000061073AC8B4C50004ED3Ad0s2
```

▼ 如何取消对所有多路径 FC 设备的配置

光纤连接的主机端口上的一个 Ap_Id 是多路径设备的一个路径。

当多路径设备连接了多个 Ap_Id 时，在您取消对 Ap_Id 的配置之后，系统仍可使用该设备。在取消最后一个 Ap_Id 的配置后，不再保留任何其他路径，因此系统无法使用该设备。只能取消配置光纤连接的主机端口上的设备。

1. 成为管理员。
2. 确定要取消配置的设备。

```
# cfmadm -al
Ap_Id      Type      Receptacle  Occupant  Condition
c0          fc-fabric connected   configured unknown
c0::50020f2300006077 disk      connected   configured unknown
c0::50020f23000063a9 disk      connected   configured unknown
c1          fc-private connected   configured unknown
c1::220203708b69c32b disk      connected   configured unknown
c1::220203708b9b20b2 disk      connected   configured unknown
c2          fc-fabric connected   configured unknown
c2::50020f2300005f24 disk      connected   configured unknown
```

3. 取消配置选定端口上的所有已配置设备。

```
# cfmadm -c unconfigure c2
```

注 - 此操作将对 c2 上的所有设备重复单个设备的 unconfigure 命令。如果 c2 上的设备数量巨大，此过程可能很耗时。

4. 检验 c2 上的所有设备是否已取消配置。

```
# cfmadm -al
Ap_Id      Type      Receptacle  Occupant  Condition
c0          fc-fabric connected   configured unknown
c0::50020f2300006077 disk      connected   configured unknown
c1          fc-private connected   configured unknown
c1::220203708b69c32b disk      connected   configured unknown
c1::220203708ba7d832 disk      connected   configured unknown
c2          fc-fabric connected   unconfigured unknown
```

```
c2::50020f2300005f24 disk      connected  unconfigured unknown
c2::50020f2300006107 disk      connected  unconfigured unknown
```

请注意，Occupant 列将 c2 和连接到 c2 的所有设备列为 unconfigured。

支持的 FC-HBA API

本附录包含受支持的 FC-HBA 接口的列表。有关 API 的更多信息，请参阅[“Solaris I/O 多路径概述” \[9\]](#)。

受支持的光纤通道 HBA API

Solaris I/O 多路径支持以下 FC-HBA 接口：

- HBA_GetVersion
- HBA_LoadLibrary
- HBA_FreeLibrary
- HBA_GetNumberOfAdapters
- HBA_GetAdapterName
- HBA_OpenAdapter
- HBA_CloseAdapter
- HBA_GetAdapterAttributes
- HBA_GetAdapterPortAttributes
- HBA_GetDiscoveredPortAttributes
- HBA_GetPortAttributesbyWWN
- HBA_SendCTPassThru
- HBA_SendCTPassThruV2
- HBA_RefreshInformation
- HBA_GetFcpTargetMapping
- HBA_SendScsiInquiry
- HBA_SendReportLuns
- HBA_SendReadCapacity
- HBA_SetRNIDMgmtInfo
- HBA_GetRNIDMgmtInfo
- HBA_SendRNID

- HBA_SendRNIDV2
- HBA_ScsiInquiryV2
- HBA_ScsiReportLUNsV2
- HBA_ScsiReadCapacityV2
- HBA_OpenAdapterByWWN
- HBA_RefreshAdapterConfiguration
- HBA_GetVendorLibraryAttributes
- HBA_GetFcpTargetMappingV2
- HBA_SendRLS
- HBA_RegisterForAdapterEvents
- HBA_RegisterForAdapterAddEvents
- HBA_RegisterForAdapterPortEvents
- HBA_RegisterForTargetEvents
- HBA_RegisterForAdapterTargetEvents

多路径相关问题故障排除

本附录提供了运行多路径功能时可能会发生的潜在问题的解决方案。

本章包含以下主题：

- [如何在单用户模式中恢复引导故障 \[125\]](#)
- [如何从系统崩溃中恢复 \[125\]](#)

在 stmsboot 期间系统引导失败或崩溃

如果系统在 stmsboot 启用 (-e)、禁用 (-d) 或更新 (-u) 操作之后无法引导，请执行以下步骤，在单用户模式中进行恢复。

▼ 如何在单用户模式中恢复引导故障

1. 键入 root 用户口令并进入单用户模式。
2. 重新启动 mpvio-upgrade 服务。

```
# svcadm restart svc:/system/device/mpvio-upgrade:default
```

如果此命令不成功，请按照下节中的说明恢复原始配置。

▼ 如何从系统崩溃中恢复

如果在 stmsboot 启用 (-e)、禁用 (-d) 或更新 (-u) 操作之后，系统挂起、出现紧急情况或无法引导，请执行以下步骤恢复原始多路径配置。

1. 从其他可引导的磁盘、安装 DVD 或网络引导系统。

- SPARC：如果从安装介质或网络上的安装服务器进行引导，请选择文本安装。如果从安装服务器进行引导，请使用以下命令：

```
ok boot net:dhcp
```

- x86：如果从安装介质或网络上的安装服务器进行引导，请从 GRUB 菜单选择以下文本安装选项：

```
Oracle Solaris 11.1 Text Installer and command line
```

- 从以下菜单选择 3 Shell 选项：

```
1 Install Oracle Solaris
2 Install Additional Drivers
3 Shell
4 Terminal type (currently sun-color)
5 Reboot
```

```
Please enter a number [1]: 3
```

```
To return to the main menu, exit the shell
```

2. 导入您的 ZFS 根池。

```
# zpool import -f rpool
```

3. 挂载根 BE。

```
# mkdir /a
# beadm mount solaris /a
```

4. 如下所示恢复原始 `fp.conf` 文件（对于 FC 多路径）或 `mpt.conf`（对于 SAS 多路径）。

- 如果您运行了 `stmsboot -e` 命令或 `stmsboot -d` 命令：

- 对于 SAS 多路径：

```
# cp /a/etc/mpxio/mpt.conf /a/etc/driver/drv/mpt.conf
```

- 对于 FC 多路径：

```
# cp /a/etc/mpxio/fp.conf /a/etc/driver/drv/fp.conf
```

- 如果您运行了 `stmsboot -u` 命令，并修改了 `fp.conf` 文件或 `mpt.conf`，请编辑 `/a/etc/driver/drv/fp.conf` 或 `/a/etc/driver/drv/mpt.conf` 文件，撤消对该文件所做的更改。

5. 在运行 `stmsboot` 命令之前，先撤消对多路径配置所做的所有其他更改。

例如，如果您修改了 `scsi_vhci.conf` 文件，则编辑 `/a/etc/driver/drv/scsi_vhci.conf` 文件，撤消对该文件所做的更改。

例如，如果您修改了 `scsi_vhci.conf` 文件，则编辑 `/a/kernel/drv/scsi_vhci.conf` 文件，撤消对该文件所做的更改。

如果您修改了存储阵列的设备设置，请恢复其原始设置。

6. 恢复原始 `/etc/vfstab` 文件：

```
# cp /a/etc/mpxio/vfstab /a/etc/vfstab
```

`/a/etc/mpxio/vfstab` 文件是 `stmsboot` 命令在更新 `vfstab` 文件之前保存的原始 `/etc/vfstab` 文件的副本。如果 `stmsboot` 命令没有修改 `vfstab` 文件，则 `/a/etc/mpxio/vfstab` 文件将不存在。

7. 如果系统在基于 x86 的系统上的 Oracle Solaris OS 上运行，请执行以下步骤：

a. 恢复原始 `/boot/solaris/bootenv.rc` 文件。

```
# cp /a/etc/mpxio/bootenv.rc /a/boot/solaris/bootenv.rc
```

`/a/etc/mpxio/bootenv.rc` 文件是 `stmsboot` 命令在更新 `bootenv.rc` 文件之前保存的原始 `/boot/solaris/bootenv.rc` 文件的副本。如果 `stmsboot` 命令没有修改 `bootenv.rc` 文件，则 `/a/etc/mpxio/bootenv.rc` 文件将不存在。

b. 更新引导归档文件。

```
# bootadm update-archive -R /a
```

8. 禁用 `mpxio-upgrade` 服务：

```
# /usr/sbin/svccfg -f /a/etc/mpxio/svccfg_recover
```

9. 卸载 BE。

```
# beadm umount solaris
```

10. 卸载 UFS 根文件系统。

```
# umount /a
```

11. 重新引导系统。

调优磁盘目标驱动程序属性

此附录介绍如何通过使用 `sd-config-list` 或 `ssd-config-list` 全局属性，在 `.conf` 文件中调优 Oracle Solaris 磁盘目标驱动程序（`sd` 或 `ssd` 驱动程序）属性。

该附录介绍了以下用于调优磁盘目标驱动程序属性的格式：

- “用于调优磁盘驱动程序的名称:值对格式” [130]
- “用于调优磁盘驱动程序的位掩码格式” [132]

磁盘驱动程序的可调参数

用于调优磁盘驱动程序的可调参数使用以下前缀类别：

- BCD – 二进制编码的十进制，如表 D-1 “不同平台上支持的可调参数及其标志值” 中所示
- `delay` – 发出重试时的延迟时间
- `timeout` – 进程允许的最大时间
- `reset` – 重置控制
- `retries` – 失败之前的重试次数
- `throttle` – 活动控制

Oracle Solaris 10 和 Oracle Solaris 11 支持以下可调参数名称及其数据类型：

<code>cache-nonvolatile</code>	BOOLEAN
<code>controller-type</code>	UINT32
<code>delay-busy</code>	UINT32
<code>disksort</code>	BOOLEAN
<code>emulation-rmw</code>	UINT32

physical-block-size	UINT32
reset-lun	BOOLEAN
retries-busy	UINT32
retries-timeout	UINT32
retries-notready	UINT32
retries-reset	UINT32
rmw-type	UINT32
timeout-releasereservation	UINT32
throttle-max	UINT32
throttle-min	UINT32

仅在 Oracle Solaris 11 中受支持的可调参数全都是 BOOLEAN 数据类型。这些参数包括：

- cdb-suppress-dpofua
- mmc-gesn-polling
- power-condition

注 - BOOLEAN 值必须为 TRUE 或 FALSE。

用于调优磁盘驱动程序的名称:值对格式

sd 和 ssd 驱动程序支持 JSON 文本名称:值格式，从而支持设置特定的可调参数属性值。

对 sd 驱动程序使用以下语法：

```
sd-config-list = duplet [, duplet]*;
```

注 - 请确保用分号作为条目的结尾，否则配置将无效，属性将保留其缺省值。

对 ssd 驱动程序使用以下语法：

```
ssd-config-list = duplet [, duplet]*;
```

其中 *duplet* 为 "VID PID", "tunable [, tunable]*"

VID	设备的供应商标识 (vendor identification, VID) 字段的内容响应 SCSI INQUIRY 命令。VID 字段长度必须为八个字符。如果 VID 字段少于八个字符, 则必须添加空格使 VID 的长度达到八个字符。请参见 例 D-1 “在一个 sd.conf 文件中配置两个目标设备” 。
PID	设备的产品标识 (product identification, PID) 字段的内容响应 SCSI INQUIRY 命令。PID 最多可以包含 16 个左对齐的字符。如果提供的字符不到 16 个, 则仅对您提供的 PID 长度进行比较。
tunable[, tunable]*	tunable 是名称:值对。

注 - 如果可调参数拼写错误或者条目不以分号结尾, 则配置无效, 设备属性将保留其缺省值。

当 SCSI INQUIRY 命令返回的前缀值与 sd-config-list 或 ssd-config-list 中的 PID 相同时, 则认为 PID 值是一个匹配项。例如, 如果 CMS200 是 sd-config-list 或 ssd-config-list 条目中的 PID, 那么, 当 SCSI INQUIRY 命令返回的 PID 为 CMS200-R、CMS200-T 或 CMS200-UV10 时, 它们都被视为匹配项。

在 SPARC 平台上, 目标设备可以绑定到 sd 或 ssd 驱动程序, 具体取决于设备是否为 FC 设备以及设备上是否启用了 MPxIO。可以使用 prtconf 命令检查设备配置信息。请参见 [prtconf\(1M\)](#) 手册页。

例 D-1 在一个 sd.conf 文件中配置两个目标设备

以下示例说明了如何在一个 sd.conf 文件中配置两个目标设备 SAMPLE 和 SUM。

```
sd-config-list = "SAMPLE STTU1234566AB", "delay-busy:6000000000",
                "SUM ABC200_R", "retries-busy:5, throttle-max:300";
```

在此示例中, 对于设备 SAMPLE, VID 为 SAMPLE, PID 为 STTU1234566AB。重试之前的延迟时间设置为 6 秒。

对于设备 SUM, VID 为 SUM, PID 为 ABC200_R。针对 I/O 忙状态的重试次数设置为 5。最大限制值设置为 300。

例 D-2 在一个 ssd.conf 文件中配置两个目标设备

以下示例说明了如何在一个 ssd.conf 文件中配置两个目标设备 GATES 和 SINE。

```
ssd-config-list = "GATES AB568536611CD46G", "reset-lun:TRUE",
                 "SINE XYZ200_R", "retries-notready:6, throttle-min:200";
```

在此示例中，对于设备 GATES，VID 为 GATES，PID 为 AB568536611CD46G。reset-lun 可调参数的值为 TRUE 时表示重置 LUN。

对于设备 SINE，VID 为 SINE，PID 为 XYZ200_R。I/O 未就绪时的重试次数设置为 6。最小限制值设置为 200。

用于调优磁盘驱动程序的位掩码格式

可以通过使用位掩码格式（也称为 Version1 格式）调优参数。位掩码格式包括 sd-config-list 属性，其中包含属性数组条目。

对 sd 驱动程序使用以下语法：

```
sd-config-list = duplet [, duplet ]*;  
sd-ver1-conf-data = 1, mask, value-sequence;
```

注 - 请确保用分号作为条目的结尾，否则配置将无效，属性将保留其缺省值。

对 ssd 驱动程序使用以下语法：

```
ssd-config-list = duplet [, duplet ]*;  
sd-ver1-conf-data = 1, mask, value-sequence;
```

其中 *duplet* 为 "VID PID", "sd-ver1-conf-data"

VID	设备的供应商标识 (vendor identification, VID) 字段的内容响应 SCSI INQUIRY 命令。VID 字段长度必须为八个字符。如果供应商标识字段少于八个字符，则必须添加空格使 VID 的长度达到八个字符。	
PID	设备的产品标识 (product identification, PID) 字段的内容响应 SCSI INQUIRY 命令。PID 最多可以包含 16 个左对齐的字符。如果提供的字符不到 16 个，则仅对您提供的 PID 长度进行比较。	
sd-ver1-conf-data	属性数组由一个版本号 1、一个掩码号和要设置的可调参数值组成。	
	mask	0x01 到 0x7FFFF 之间的值。mask 的值有时可以是多个所需标志的 OR 运算结果。 表 D-1 “不同平台上支持的可调参数及其标志值” 显示了每个可调参数的位值。
	value-sequence	对应于 mask 值的有效属性值序列和 0 的数量。value-sequence 的长度限制为 19 个字符。有关可调参数的数据类型，请参见 “磁盘驱动程序的可调参数” [129] 。

注 - 具有不同 VID 和 PID 的 duplet 共享相同的 *sd-ver1-conf-data* 属性数组，或者定义自己的属性数组。可以定制属性数组名称。

当 SCSI INQUIRY 命令返回的前缀值与 *sd-config-list* 或 *ssd-config-list* 中的 PID 相同时，则认为 PID 值是一个匹配项。例如，如果 CMS200 是 *sd-config-list* 或 *ssd-config-list* 条目中的 PID，那么，当 SCSI INQUIRY 命令返回的 PID 为 CMS200-R、CMS200-T 或 CMS200-UV10 时，它们都被视为匹配项。

每个位的位置定义取决于平台。可调参数可能对应于不同平台上的不同标志值。Oracle Solaris 11 和 Oracle Solaris 10 支持采用位掩码格式的由 19 个可调参数组成的相同集合。

下表列出了不同平台上的可调参数及其标志值。

表 D-1 不同平台上支持的可调参数及其标志值

位	标志值	SPARC 上的 sd 驱动程序	SPARC 上的 ssd 驱动程序	x86 或 x64 上的 sd 驱动程序
1	0x00001	throttle-max	throttle-max	throttle-max
2	0x00002	controller-type	retries-notready	controller-type
3	0x00004	retries-notready	retries-busy	fab-devid
4	0x00008	fab-devid	fab-devid	disable_caching
5	0x00010	disable_caching	disable_caching	BCD-play
6	0x00020	retries-busy	controller-type	BCD-read-subchannel
7	0x00040	BCD-play	BCD-play	BCD-read-TOC-TRK
8	0x00080	BCD-read-subchannel	BCD-read-subchannel	BCD-read-TOC-ADDR
9	0x00100	BCD-read-TOC-TRK	BCD-read-TOC-TRK	no-READ-HDR
10	0x00200	BCD-read-TOC-ADDR	BCD-read-TOC-ADDR	CD-read-XD4
11	0x00400	no-READ-HDR	no-READ-HDR	retries-notready
12	0x00800	CD-read-XD4	CD-read-XD4	retries-busy
13	0x01000	retries-reset	retries-reset	retries-reset
14	0x02000	timeout-releasereservation	timeout-releasereservation	timeout-releasereservation
15	0x04000	tur-check	tur-check	tur-check
16	0x08000	throttle-min	throttle-min	throttle-min
17	0x10000	disksort	disksort	disksort
18	0x20000	reset-lun	reset-lun	reset-lun
19	0x40000	cache-nonvolatile	cache-nonvolatile	cache-nonvolatile

例 D-3 在 x86 平台上配置 sd 驱动程序的可调参数

此示例说明了如何在 x86 平台上调优 *sd.conf* 文件中参数。

索引

数字和符号

24 位 FC 寻址设备, 108

A

ANSI 标准多路径管理 API, 27

Ap_Id, 109

C

磁带设备的持久绑定, 103, 103

存储设备, 26

错误消息

iSCSI, 72

cfgadm

-c, 110

-l, 109, 110

cfgadm -al -o show_SCSI_LUN, 108

cfgadm -c unconfigure, 118

cfgadm configure, 110

cfgadm_fp(1M), 113

cfgadm(1M), 109

D

多路径

mpathadm 命令, 27

SAS 引导设备配置, 88

SAS 注意事项, 87

光纤设备节点配置, 108

功能, 11

启用和禁用, 18

基于每端口启用和禁用, 21

已禁用, 多个设备配置, 111

已禁用, 设备节点配置, 110

引导注意事项, 45

手动配置, 107

支持, 27

故障排除, 125

配置第三方存储设备, 23

配置自动故障恢复, 26

配置设备节点, 112

多路径, 启用和禁用, 18

多路径管理 API 插件

属性, 27

库, 27

F

访问

iSCSI 磁盘 (如何), 52

fc-fabric, 113

FCAL, 10

FCoE

创建端口, 82

删除端口, 83

报告端口状态, 83

配置端口, 82

限制, 81

G

故障排除

iSCSI 配置问题 (如何), 70

多路径, 125

光纤连接的主机端口, 109, 113

光纤设备配置, 110

光纤通道

HBA, 10

IPFC 注意事项, 89

仲裁环路, 10

- 列出 LUN 级别信息, 108
- 受支持的 HBA 接口, 123
- 启用和禁用多路径, 18
- 多路径配置, 17
- 引导基于 x86 的系统, 95
- 确定适配器端口实例, 90
- 配置 IPFC SAN 设备, 89
- 光纤通道端口
 - 配置, 77
- 广播, 89

H

- 混杂模式, 89

I

- IPFC
 - 注意事项, 89
- iSCSI
 - 一般 iSCSI 错误消息, 72
 - 修改 iSCSI 启动器和目标参数 (如何), 67
 - 删除搜索的 iSCSI 目标 (如何), 53
 - 动态和静态目标搜索, 50
 - 监视 iSCSI 配置 (如何), 62
 - 解决 iSCSI 配置问题 (如何), 70
 - 访问 iSCSI 磁盘 (如何), 52
 - 配置单向或双向 CHAP 验证 (如何), 55
- iscsiadm list
 - 显示 iSCSI 配置信息 (示例), 63
- iscsiadm modify 命令
 - 启用 CHAP (示例), 56
 - 启用或禁用静态或动态目标 (示例), 53
- iscsiadm remove 命令
 - 删除静态或动态目标 (示例), 54

J

- 基于 x86 的系统 OS 引导过程
 - 光纤通道, 95
- 基于光纤通道的 Internet 协议 见 IPFC
- 监视
 - iSCSI 配置 (如何), 62
- 禁用
 - 多路径, 18

L

- LUN
 - SCSI 设备, 115
 - 屏蔽, 43
 - 识别, 44
- luxadm(1M), 27

M

- 每端口配置
 - 启用和禁用多路径, 21
 - 注意事项, 21
- 名称:值格式
 - JSON 文本格式, 130
- modinfo, 109
- mpathadm 命令, 27
 - 具有特定名称的逻辑单元, 33
 - 启动器端口
 - 属性, 28
 - 启用路径, 39
 - 禁用路径, 41
 - 自动故障恢复, 配置, 35
 - 逻辑单元
 - 与目标端口相关联, 31
 - 属性和关联路径, 29
 - 故障转移, 36
 - 配置自动故障恢复, 35
- mpt 驱动程序, 87

N

- NPIV 端口
 - 配置, 77

P

- 配置
 - dump, 18
 - FC 设备, 43
 - FCoE, 82
 - iSCSI 的单向或双向 CHAP 验证 (如何), 55
 - NPIV 端口, 77
 - SAS 设备, 88, 88
 - 任务, 14
 - 光纤设备, 110

- 光纤连接设备，43
- 多路径，17
- 多路径功能，17，43
- 手动，107
- 第三方设备，23，24
- 自动故障恢复，26
- 虚拟光纤通道端口，77
- 设备，109

Q

- 启动器端口
 - 显示属性，28
- 启用
 - 多路径，18

R

- REPORT LUNS 命令，23

S

- 删除
 - 搜索的 iSCSI 目标（如何），53
- 设备
 - 存储，12
 - 配置，110
 - 非对称，12
- 设备, 持久性命名
 - 磁带，9
- 设备管理，14
- 搜索
 - SAS 设备，87
 - 动态，87
- SAS 设备
 - 动态搜索，87
 - 多路径注意事项，87
 - 配置，88
- sd 驱动程序，87
- show_SCSI_LUN，111，115
- snoop(1M)，89
- ssd 驱动程序，109
- st 驱动程序，109
- stmsboot 命令，18
 - FC 设备注意事项，43

- 启用和禁用多路径，12
- 引导注意事项，45
- 注意事项，15
- 确定设备路径，18
- 系统崩溃，125

T

- T10 标准，27
- T11 标准，10

W

- 位掩码格式
 - Version1 格式，132
- 物理设备，111

X

- 限制
 - FCoE，81
- 修改
 - iSCSI 启动器和目标参数（如何），67

Y

- 引导，10
- 引导磁盘，45

Z

- 注意事项
 - IPFC，89
 - SAN 设备，43
 - SAS 设备，87
 - stmsboot 命令，15
 - StorageTek Traffic Manager，15
 - 光纤引导设备，44
 - 多路径，17
 - 每端口配置，21
 - 特定于设备，17
 - 第三方设备配置，23
 - 设备名称更改，17
- 自动故障恢复
 - 配置，26

