

Oracle® Solaris 11.1 可调参数参考手册

版权所有 © 2000, 2013, Oracle 和/或其附属公司。保留所有权利。

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

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

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

U.S. GOVERNMENT END USERS:

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

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

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

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

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

目录

前言	13
1 Oracle Solaris 系统调优概述	17
Oracle Solaris 系统调优中有哪些新增内容?	17
调优 Oracle Solaris 系统	18
可调参数说明的调优格式	19
调优 Oracle Solaris 内核	20
/etc/system 文件	21
kldb 命令	21
mdb 命令	22
特殊的 Oracle Solaris tune 结构和 var 结构	22
查看 Oracle Solaris 系统配置信息	23
sysdef 命令	23
kstat 实用程序	23
2 Oracle Solaris 内核可调参数	25
何处查找可调参数信息	25
常规内核及内存参数	26
physmem	26
default_stksize	26
lwp_default_stksize	27
logevent_max_q_sz	28
segkpsize	28
noexec_user_stack	29
fsflush 和相关参数	30
fsflush	30
tune_t_fsflushr	30

autoup	31
dopageflush	32
doiflush	32
进程大小调整参数	33
maxusers	33
reserved_procs	34
pidmax	34
max_nprocs	35
maxuprc	36
ngroups_max	36
分页相关参数	37
lotsfree	38
desfree	39
minfree	40
throttlefree	40
pageout_reserve	41
pages_pp_maximum	42
tune_t_minarmem	42
fastscan	43
slowscan	43
min_percent_cpu	44
handspreadpages	44
pages_before_pager	45
maxpgio	45
与交换相关的参数	46
swapfs_reserve	46
swapfs_minfree	47
内核内存分配器	47
kmem_flags	48
kmem_stackinfo	49
常规驱动程序参数	50
moddebug	50
ddi_msix_alloc_limit	51
网络驱动程序参数	51
igb 参数	51
ixgbe 参数	52

常规 I/O 参数	55
maxphys	55
rlim_fd_max	56
rlim_fd_cur	57
常规文件系统参数	57
ncsize	57
dnlc_dir_enable	58
dnlc_dir_min_size	58
dnlc_dir_max_size	59
dnlc_dircache_percent	59
TMPFS 参数	60
tmpfs:tmpfs_maxkmem	60
tmpfs:tmpfs_minfree	61
伪终端	61
pt_cnt	62
pt_pctofmem	63
pt_max_pty	63
STREAMS 参数	64
nstrpush	64
strmsgsz	64
strctlsz	65
System V 消息队列	65
System V 信号	65
System V 共享内存	66
segspt_minfree	66
调度	66
disp_rechoose_interval	66
计时器	68
hires_tick	68
timer_max	68
特定于 SPARC 系统的参数	68
tsb_alloc_hiwater_factor	69
default_tsb_size	69
enable_tsb_rss_sizing	70
tsb_rss_factor	70
地址组参数	71

lpg_alloc_prefer	71
lgrp_mem_pset_aware	72
3 Oracle Solaris ZFS 可调参数	73
何处查找可调参数信息	73
ZFS 调优的注意事项	73
ZFS ARC 参数	74
zfs_arc_min	74
zfs_arc_max	74
ZFS 文件级预取	75
zfs_prefetch_disable	75
ZFS 设备 I/O 队列深度	76
zfs_vdev_max_pending	76
使用闪存存储时调优 ZFS	77
将闪存设备添加为 ZFS 日志设备或高速缓存设备	77
确保闪存设备和 NVRAM 存储设备具有正确的高速缓存刷新行为	78
为数据库产品进行 ZFS 调优	80
为 Oracle 数据库进行 ZFS 调优	81
ZFS 与 MySQL 结合使用的注意事项	84
4 NFS 可调参数	85
何处查找可调参数信息	85
调优 NFS 环境	85
NFS 模块参数	86
nfs:nfs3_pathconf_disable_cache	86
nfs:nfs4_pathconf_disable_cache	86
nfs:nfs_allow_preepoch_time	87
nfs:nfs_cots_timeo	87
nfs:nfs3_cots_timeo	88
nfs:nfs4_cots_timeo	88
nfs:nfs_do_symlink_cache	89
nfs:nfs3_do_symlink_cache	89
nfs:nfs4_do_symlink_cache	90
nfs:nfs_dynamic	90
nfs:nfs3_dynamic	91

nfs:nfs_lookup_neg_cache	91
nfs:nfs3_lookup_neg_cache	92
nfs:nfs4_lookup_neg_cache	93
nfs:nfs_max_threads	93
nfs:nfs3_max_threads	94
nfs:nfs4_max_threads	95
nfs:nfs_nra	95
nfs:nfs3_nra	96
nfs:nfs4_nra	96
nfs:nrnode	97
nfs:nfs_shrinkreaddir	98
nfs:nfs3_shrinkreaddir	98
nfs:nfs_write_error_interval	99
nfs:nfs_write_error_to_cons_only	100
nfs:nfs_disable_rddir_cache	100
nfs:nfs3_bsize	101
nfs:nfs4_bsize	101
nfs:nfs_async_clusters	102
nfs:nfs3_async_clusters	103
nfs:nfs4_async_clusters	103
nfs:nfs_async_timeout	104
nfs:nacache	105
nfs:nfs3_jukebox_delay	105
nfs:nfs3_max_transfer_size	106
nfs:nfs4_max_transfer_size	107
nfs:nfs3_max_transfer_size_clts	107
nfs:nfs3_max_transfer_size_cots	108
rpcmod 模块参数	108
rpcmod:clnt_max_conns	108
rpcmod:clnt_idle_timeout	109
rpcmod:svc_idle_timeout	109
rpcmod:svc_default_stksize	110
rpcmod:maxdupreqs	110
rpcmod:cotsmaxdupreqs	111

5 Internet 协议套件可调参数	113
何处查找可调参数信息	113
调优 IP 套件参数的概述	113
IP 套件参数验证	114
Internet 请求注解 (RFC)	114
IP 可调参数	114
_icmp_err_interval and _icmp_err_burst	114
_respond_to_echo_broadcast 和 _respond_to_echo_multicast (ipv4 或 ipv6) ..	115
_send_redirects (ipv4 或 ipv6)	115
forwarding (ipv4 或 ipv6)	116
ttl	116
hoplimit (ipv6)	116
_addr_per_if	117
hostmodel (ipv4 或 ipv6)	117
ip_queue_fanout	118
需要额外注意的 IP 可调参数	118
TCP 可调参数	119
_deferred_ack_interval	119
_local_dack_interval	120
_deferred_acks_max	120
_local_dacks_max	121
_wscale_always	121
_tstamp_always	122
send_buf	122
recv_buf	123
max_buf	123
_cwnd_max	123
_slow_start_initial	124
_local_slow_start_initial	124
_slow_start_after_idle	125
sack	125
_rev_src_routes	126
_time_wait_interval	126
ecn	126
_conn_req_max_q	127
_conn_req_max_q0	128

_conn_req_min	128
_rst_sent_rate_enabled	129
_rst_sent_rate	129
在 /etc/system 文件中设置的 TCP/IP 参数	131
需要额外注意的 TCP 参数	132
UDP 可调参数	135
send_buf	135
recv_buf	136
max_buf	136
smallest_anon_port	137
largest_anon_port	137
IPQoS 可调参数	138
_policy_mask	138
SCTP 可调参数	138
_max_init_retr	138
_pa_max_retr	139
_pp_max_retr	139
_cwnd_max	139
_ipv4_ttl	140
_heartbeat_interval	140
_new_secret_interval	141
_initial_mtu	141
_deferred_ack_interval	142
_ignore_path_mtu	142
_initial_ssthresh	142
send_buf	143
_xmit_lowat	143
recv_buf	143
max_buf	144
_rto_min	144
_rto_max	145
_rto_initial	145
_cookie_life	145
_max_in_streams	146
_initial_out_streams	146
_shutack_wait_bound	146

_maxburst	147
_addip_enabled	147
_prsctp_enabled	147
smallest_anon_port	148
largest_anon_port	148
每路由度量	149
6 系统工具参数	151
系统缺省参数	152
autofs	152
cron	152
devfsadm	152
dhcpgent	152
fs	153
ftp	153
inetinit	153
init	153
ipsec	154
kbd	154
keyserv	154
login	154
mpathd	154
nfs	155
nfslogd	155
nss	155
passwd	155
su	155
syslog	155
tar	155
telnetd	156
utmpd	156
A 可调参数变更历史记录	157
内核参数	157
常规内核及内存参数（Oracle Solaris 11 和 Oracle Solaris 11.1）	157

分页相关参数 (Oracle Solaris 11)	158
进程大小调整可调参数 (Oracle Solaris 11.1)	158
与交换相关的参数 (Oracle Solaris 11.1)	159
常规文件系统参数 (Oracle Solaris 11.1)	159
常规驱动程序参数 (Oracle Solaris 11)	159
网络驱动程序参数 (Oracle Solaris 11)	159
TCP/IP 可调参数 (Oracle Solaris 11 和 Oracle Solaris 11.1)	160
[tcp,sctp,udp]_smallest_anon_port 和 [tcp,sctp,udp]_largest_anon_port	160
_local_slow_start_initial	160
IP 参数名变更 (Oracle Solaris 11)	160
TCP 参数名变更 (Oracle Solaris 11)	161
UDP 参数名变更 (Oracle Solaris 11)	163
SCTP 参数名变更 (Oracle Solaris 11)	163
已过时或已删除的参数 (Oracle Solaris 11 和 Oracle Solaris 11.1)	164
NCA 参数	165
consistent_coloring	165
rstchown	165
已过时的 TCP/IP 模块参数 (Oracle Solaris 11)	165
B 本手册的修订历史记录	167
当前版本: Oracle Solaris 11.1 发行版	167
Oracle Solaris 发行版中新增或已更改的参数	167
 索引	 169

前言

《Oracle Solaris 11.1 可调参数参考手册》提供了关于 Oracle Solaris OS 内核和网络可调参数的参考信息。本手册没有提供关于桌面系统或 Java 环境的可调参数信息。

本手册同时包含了适用于基于 SPARC 和基于 x86 的系统的信息。

注 - 此 Oracle Solaris 发行版支持使用 SPARC 和 x86 系列处理器体系结构的系统。支持的系统可以在 <http://www.oracle.com/webfolder/technetwork/hcl/index.html> 上的 Oracle Solaris Hardware Compatibility List (Oracle Solaris 硬件兼容性列表) 中找到。本文档列举了在不同类型的平台上进行实现时的所有差别。

目标读者

本书适用于经验丰富的 Oracle Solaris 系统管理员，他们在特定情况下可能需要更改内核可调参数。有关更改 Oracle Solaris 可调参数的指南，请参见第 18 页中的“[调优 Oracle Solaris 系统](#)”。

本书的结构

下表介绍了本书中的各章和附录。

章	说明
第 1 章, Oracle Solaris 系统调优概述	有关调优 Oracle Solaris 系统的概述。还提供了本书中用来描述内核可调参数的格式的说明。
第 2 章, Oracle Solaris 内核可调参数	对 Oracle Solaris 内核可调参数 (例如内核内存、文件系统、进程大小和分页参数) 的说明。
第 3 章, Oracle Solaris ZFS 可调参数	Oracle Solaris ZFS 可调参数的说明。本章还包括数据库产品的 ZFS 调优信息。
第 4 章, NFS 可调参数	对 NFS 可调参数 (例如缓存符号链接、动态重新传输和 RPC 安全性参数) 的说明。

章	说明
第 5 章, Internet 协议套件可调参数	对 TCP/IP 可调参数（例如 IP 转发、源路由选择和缓冲区大小调整参数）的说明。
第 6 章, 系统工具参数	对用于为特定系统工具设置缺省值的参数的说明。更改是通过修改 <code>/etc/default</code> 目录中的文件执行的。
附录 A, 可调参数变更历史记录	已更改或现在已过时的参数的历史记录。
附录 B, 本手册的修订历史记录	本手册的修订历史记录, 包括当前 Oracle Solaris 发行版。

关于 Oracle Solaris 调优信息的其他资源

该表介绍了关于 Oracle Solaris 调优信息的其他资源。

调优资源	更多信息
深度技术白皮书	http://www.oracle.com/technetwork/server-storage/solaris/overview/index.html

获取 Oracle 支持

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

印刷约定

下表介绍了本书中的印刷约定。

表 P-1 印刷约定

字体或符号	含义	示例
AaBbCc123	命令、文件和目录的名称; 计算机屏幕输出	编辑 <code>.login</code> 文件。 使用 <code>ls -a</code> 列出所有文件。 <code>machine_name% you have mail.</code>
AaBbCc123	用户键入的内容, 与计算机屏幕输出的显示不同	<code>machine_name% su</code> <code>Password:</code>

表 P-1 印刷约定 (续)

字体或符号	含义	示例
<i>aabbcc123</i>	要使用实名或值替换的命令行占位符	删除文件的命令为 <i>rm filename</i> 。
AaBbCc123	保留未译的新词或术语以及要强调的词	这些称为 <i>Class</i> 选项。 注意： 有些强调的项目在联机时以粗体显示。
新词术语强调	新词或术语以及要强调的词	高速缓存 是存储在本地的副本。 请勿保存文件。
《书名》	书名	阅读《用户指南》的第 6 章。

命令中的 shell 提示符示例

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

表 P-2 shell 提示符

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

Oracle Solaris 系统调优概述

本节提供了有关本手册中调优信息的格式的概述信息。本节还介绍了调优 Oracle Solaris 系统的不同方式。

- 第 17 页中的“Oracle Solaris 系统调优中有哪些新增内容？”
- 第 18 页中的“调优 Oracle Solaris 系统”
- 第 19 页中的“可调参数说明的调优格式”
- 第 20 页中的“调优 Oracle Solaris 内核”
- 第 22 页中的“特殊的 Oracle Solaris tune 结构和 var 结构”
- 第 23 页中的“查看 Oracle Solaris 系统配置信息”
- 第 23 页中的“kstat 实用程序”

Oracle Solaris 系统调优中有哪些新增内容？

本节介绍了 Oracle Solaris 11.1 发行版中新增或已更改的参数。

- Oracle Solaris 11.1：Oracle Solaris ZFS 可调参数信息位于第 3 章，Oracle Solaris ZFS 可调参数。
- Oracle Solaris 11.1：maxusers、max_nprocs、ngroups_max、pidmax 和 segkpsize 参数已修改。有关更多信息，请参见第 2 章，Oracle Solaris 内核可调参数。
- Oracle Solaris 11.1：dnlc_dircache_percent 参数是新增参数。有关更多信息，请参见第 59 页中的“dnlc_dircache_percent”。
- Oracle Solaris 11：之前在 /etc/system 文件中设置的 rstchown 参数已过时。如果在 /etc/system 文件中设置此参数，将显示以下错误消息：

```
sorry, variable 'rstchown' is not defined in the 'kernel'
```

此参数已由 ZFS rstchown 文件系统属性以及一个常规文件系统挂载选项所取代。有关更多信息，请参见《Oracle Solaris 11.1 管理：ZFS 文件系统》和 mount(1M)。

- Oracle Solaris 11：以前需要通过编辑 /etc/default 目录中的文件进行配置的下列系统配置参数已更改为 SMF 服务：

- autofs
- init
- kbd
- nfs

有关更改 SMF 服务属性的信息，请参见第 6 章，[系统工具参数](#)。

- Oracle Solaris 11：ipadm 命令替换了用于设置网络属性的 ndd 命令。TCP、IP、UDP 和 SCTP 属性按以下方式进行设置：

```
ipadm set-prop -p parameter ip|ipv4|ipv6|tcp|udp|sctp
```

此外，多数网络可调参数也做了轻微变更，以便更好地关联到 ipadm 格式。

有关更多信息，请参见第 113 页中的“[调优 IP 套件参数的概述](#)”。

- Oracle Solaris 11：本发行版包括 disp_rechoose_interval 参数。有关更多信息，请参见第 66 页中的“[disp_rechoose_interval](#)”。
- Oracle Solaris 11：本发行版包括 ngroups_max 参数说明。有关更多信息，请参见第 36 页中的“[ngroups_max](#)”。
- Oracle Solaris 11：本发行版包括 zfs_arc_min 和 zfs_arc_max 参数说明。有关更多信息，请参见第 74 页中的“[zfs_arc_min](#)”和第 74 页中的“[zfs_arc_max](#)”。
- Oracle Solaris 11：本发行版包括多个 igb 和 ixgbe 网络驱动程序参数。有关更多信息，请参见第 51 页中的“[igb 参数](#)”和第 52 页中的“[ixgbe 参数](#)”。
- Oracle Solaris 11：本发行版包括了 ddi_msix_alloc_limit 参数，它可用于增大设备实例可以分配的 MSI-X 中断次数。有关更多信息，请参见第 51 页中的“[ddi_msix_alloc_limit](#)”。
- Oracle Solaris 11：本发行版包括 kmem_stackinfo 参数，启用该参数可监视内核线程栈使用情况。有关更多信息，请参见第 49 页中的“[kmem_stackinfo](#)”。
- Oracle Solaris 11：本发行版中提供了内存地址组参数。有关这些参数的更多信息，请参见第 71 页中的“[地址组参数](#)”。

调优 Oracle Solaris 系统

Oracle Solaris OS 是在 SPARC 和 x86 处理器上运行的多线程、可扩展的 UNIX 操作系统。它会根据系统负载自行调整，并且基本不需要进行调优。但是，在某些情况下，调优很有必要。本书提供了有关可用于 Oracle Solaris OS 的正式受支持的内核调优选项的详细信息。

Solaris 内核包括一个始终加载的核心部分和一系列在被引用时才加载的可加载模块。本指南的内核部分中引用的许多变量均位于核心部分中。不过，也有部分变量位于可加载模块中。

有关系统调优的一个关键注意事项是：设置系统参数（或系统变量）通常是可用来改进性能的操作中效果最差的。更改应用程序的行为通常是最有效的可用调优协助方

法。添加更多物理内存以及平衡磁盘 I/O 模式也很有用。在极少数情况下，更改本指南中所述的变量之一会对系统性能产生重大影响。

请记住，一个系统的 `/etc/system` 设置可能全部或部分不适用于另一个系统的环境。请根据变量值将应用到的环境仔细考虑该文件中的值。在尝试向此处描述的系统变量应用更改之前，请确保您已了解系统的行为。

在迁移到新的 Oracle Solaris 发行版时，建议从空的 `/etc/system` 文件开始。作为第一步，仅添加内部或第三方应用程序必需的可调参数。在建立基线测试后，评估系统性能来确定是否需要其他可调设置。



注意 - 对于不同的 Oracle Solaris 发行版，本书中介绍的可调参数可有所不同，并且确实存在不同。在发布这些可调参数后，可调参数及其说明也可能会变更，恕不另行通知。

可调参数说明的调优格式

每个可调参数说明的格式如下所示：

- 参数名
- 说明
- 数据类型
- 缺省值
- 范围
- 单位
- 动态?
- 验证
- 隐式
- 何时更改
- 区域配置
- 交付级别
- 变更历史记录

参数名

是在 `/etc/system` 文件中键入的或者从 `/etc/default/facility` 文件中找到的确切名称。

大多数参数名为 *parameter* 形式，其中不包含冒号 (:)。这些名称引用内核的核心部分中的变量。如果名称包含冒号，则冒号左侧的字符表示可加载模块的名称。参数在模块内的名称由冒号右侧的字符组成。例如：

module_name:variable

说明

简要描述参数的用途或所控制的内容。

数据类型	表示带符号或无符号的短整数或长整数。长整数的宽度是整数的两倍（以位为单位）。例如，无符号整数为 32 位，无符号长整数为 64 位。
单位	（可选）描述单位类型。
缺省值	系统将用作缺省值的值。
范围	指定系统验证或数据类型边界所允许的可能范围。 <ul style="list-style-type: none"> ■ MAXINT—有符号整数的最大值 (2,147,483,647) 的速记说明 ■ MAXUINT—无符号整数的最大值 (4,294,967,295) 的速记说明
动态?	如果可在运行系统上使用 <code>mdb</code> 或 <code>kldb</code> 调试器更改此参数，则为“是”。如果此参数仅可在引导时初始化，则为“否”。
验证	检查系统是否按 <code>/etc/system</code> 文件中的指定应用变量值或者应用变量缺省值，以及何时应用验证。
隐式	（可选）提供参数上可能存在的未阐明约束，尤其是相对于其他参数。
何时更改	解释为什么某些用户可能要更改此值。包括错误消息或返回代码。
区域配置	确定是可以在专用 IP 区域中设置此参数，还是必须在全局区域中进行设置。任何参数都不可以在共享 IP 区域中设置。
交付级别	确定界面的稳定性。本手册中的许多参数仍处于不断发展中，因此将其归类为不稳定的。有关更多信息，请参见 attributes(5) 。
变更历史记录	（可选）如适用，则包含“变更历史记录”附录的链接。

调优 Oracle Solaris 内核

下面的表格描述了可用来应用可调参数的不同方式。

以下列方式应用可调参数	更多信息
修改 <code>/etc/system</code> 文件	第 21 页中的“ <code>/etc/system</code> 文件”
使用内核调试器 (<code>kldb</code>)	第 21 页中的“ <code>kldb</code> 命令”
使用模块调试器 (<code>mdb</code>)	第 22 页中的“ <code>mdb</code> 命令”
使用 <code>ipadm</code> 命令设置 TCP/IP 参数	第 5 章，Internet 协议套件可调参数
修改 <code>/etc/default</code> 文件	第 6 章，系统工具参数

/etc/system 文件

/etc/system 文件中提供了用于调整内核参数值的静态机制。在引导时读取此文件中指定的值，然后进行应用。在重新引导系统之前，对文件所做的任何更改都不会应用到操作系统。

在计算配置参数之前，会进行一遍检查来设置所有值。

示例一 在 /etc/system 中设置参数

以下 /etc/system 条目将 ZFS ARC 最大值 (zfs_arc_max) 设置为 30 GB。

```
set zfs:zfs_arc_max = 0x78000000
```

从不正确值进行恢复

在修改 /etc/system 文件之前请创建其副本，以便值不正确时可以轻松地恢复。例如：

```
# cp /etc/system /etc/system.good
```

如果 /etc/system 文件中指定的值导致系统无法引导，则可以通过以下命令进行恢复：

```
ok boot -a
```

此命令会导致系统在引导过程中询问使用的各个文件的名称。按回车键来接受缺省值，直到要求输入 /etc/system 文件的名称。当显示

Name of system file [/etc/system]: 提示时，键入正确的 /etc/system 文件的名称或 /dev/null：

```
Name of system file [/etc/system]: /etc/system.good
```

如果指定了 /dev/null，则该路径会导致系统尝试从 /dev/null 读取它的配置信息。由于此文件是空的，因此系统将使用缺省值。引导系统后，可更正 /etc/system 文件。

有关系统恢复的更多信息，请参见《Oracle Solaris 管理：常见任务》。

kmdb 命令

kmdb 是使用与 mdb 相同的常规语法的交互式内核调试器。交互式内核调试器的一个优点是您可以设置断点。当到达断点时，您可以检查数据或单步通过内核代码的执行。

可根据需要加载及卸载 kmdb。与使用 kadb 的情况相同，执行交互式内核调试时无需重新引导系统。

有关更多信息，请参见 [kmdb\(1\)](#)。

mdb 命令

模块化调试器 `mdb` 是各个 Solaris 调试器中比较独特的，因为它可以轻松扩展。有一个编程 API 可用，可用来编辑模块以便在调试器的上下文中执行所需任务。

`mdb` 还包括一系列所需的可用性功能，包括命令行编辑、命令历史记录、内置输出分页程序、语法检查和命令流水线作业。`mdb` 是适用于内核的推荐用事后分析调试器。

有关更多信息，请参见 [mdb\(1\)](#)。

示例—使用 `mdb` 显示信息。

显示系统的内存使用情况的高级别视图。例如：

```
# mdb -k
Loading modules: [ unix genunix specfs dtrace mac cpu.generic cpu_ms.AuthenticAMD.15
uppc pcplusmp scsi_vhci zfs mpt sd ip hook neti arp usba sockfs kssl qlc fctl stmf stmf_
sbd md lofs random idm fcp crypto cpc smbsrv nfs fcip spps ufs logindmux ptm nsmb scu
mpt_sas pmcs emlxs ]
> ::memstat
Page Summary          Pages          MB  %Tot
-----
Kernel                160876          628   16%
ZFS File Data         303401         1185   30%
Anon                   25335           98    2%
Exec and libs          1459            5    0%
Page cache             5083            19    1%
Free (cachelist)      6616            25    1%
Free (freelist)       510870         1995   50%

Total                  1013640         3959
Physical               1013639         3959
> $q
```

有关使用该模块化调试器的更多信息，请参见《[Oracle Solaris Modular Debugger Guide](#)》。

在使用 `kmdb` 或 `mdb` 调试器时，模块名称前缀不是必需的。加载模块之后，其符号连同核心内核符号及之前加载的任何其他模块符号会构成一个通用名称空间。

特殊的 Oracle Solaris tune 结构和 var 结构

Oracle Solaris 可调参数表现为各种形式。`/usr/include/sys/tuneable.h` 文件中定义的 `tune` 结构是 `tune_t_fsflushr`、`tune_t_minarmem` 和 `tune_t_flkrec` 的运行时表示形式。在初始化内核后，可在 `tune` 结构的相应字段中找到对这些变量的所有引用。

在引导时为此结构设置参数的正确方法是初始化与所需字段名称对应的特殊参数。然后，系统初始化进程会将这些值加载到 `tune` 结构中。

用于放置各种可调参数的另一种结构是名为 `v` 的 `var` 结构。可以在 `/usr/include/sys/var.h` 文件中找到 `var` 结构的定义。`autoup` 和 `bufhwm` 等变量的运行时表示形式也存储在该文件中。

请不要在正在运行的系统上更改 `tune` 或 `v` 结构。在正在运行的系统上更改这些结构的任何字段都可能会导致系统故障。

查看 Oracle Solaris 系统配置信息

有多个工具可用于检查系统配置信息。某些工具要求具有超级用户特权。其他工具可由非特权用户运行。通过在正在运行的系统上使用 `mdb` 或通过 `kadb` 下引导，可以使用内核调试器检查每个结构和数据项。

有关更多信息，请参见 `mdb(1)` 或 `kadb(1M)`。

sysdef 命令

`sysdef` 命令提供了内存和进程资源限制值，以及 `tune` 和 `v` 结构的各个部分。例如，具有 500 GB 内存的 SPARC T3-4 系统上的 `sysdef`“可调参数”部分如下所示：

```
2206203904    maximum memory allowed in buffer cache (bufhwm)
65546        maximum number of processes (v.v_proc)
99           maximum global priority in sys class (MAXCLSYSPRI)
65541        maximum processes per user id (v.v_maxup)
30           auto update time limit in seconds (NAUTOUP)
25           page stealing low water mark (GPGSLO)
1            fsflush run rate (FSFLUSHR)
25           minimum resident memory for avoiding deadlock (MINARMEM)
25           minimum swapable memory for avoiding deadlock (MINASMEM)
```

有关更多信息，请参见 `sysdef(1M)`。

kstat 实用程序

`kstat` 是由各种内核子系统和驱动程序维护的数据结构。它们提供了用于将数据从内核导出到用户程序的机制，而不要求程序读取内核内存或具有超级用户特权。有关更多信息，请参见 `kstat(1M)` 或 `kstat(3KSTAT)`。

Oracle Solaris 内核可调参数

本章介绍了大多数 Oracle Solaris 内核可调参数。

- 第 26 页中的“常规内核及内存参数”
- 第 30 页中的“fsflush 和相关参数”
- 第 33 页中的“进程大小调整参数”
- 第 37 页中的“分页相关参数”
- 第 46 页中的“与交换相关的参数”
- 第 47 页中的“内核内存分配器”
- 第 50 页中的“常规驱动程序参数”
- 第 51 页中的“网络驱动程序参数”
- 第 55 页中的“常规 I/O 参数”
- 第 57 页中的“常规文件系统参数”
- 第 60 页中的“TMPFS 参数”
- 第 61 页中的“伪终端”
- 第 64 页中的“STREAMS 参数”
- 第 65 页中的“System V 消息队列”
- 第 65 页中的“System V 信号”
- 第 66 页中的“System V 共享内存”
- 第 66 页中的“调度”
- 第 68 页中的“计时器”
- 第 68 页中的“特定于 SPARC 系统的参数”
- 第 71 页中的“地址组参数”

何处查找可调参数信息

可调参数	参考
Oracle Solaris ZFS 可调参数	第 3 章, Oracle Solaris ZFS 可调参数
NFS 可调参数	第 4 章, NFS 可调参数

可调参数	参考
Internet 协议套件可调参数	第 5 章, Internet 协议套件可调参数

常规内核及内存参数

本节介绍了与物理内存和栈配置相关的常规内核参数。与 ZFS 相关的内存参数已移至第 3 章, [Oracle Solaris ZFS 可调参数](#)。

physmem

说明	在考虑 Oracle Solaris OS 和固件后, 修改系统的物理内存页数配置。
数据类型	无符号长整数
缺省值	系统上可用物理内存的可用页数, 不包括用于存储核心内核和数据的内存
范围	1 - 系统上的物理内存量
单位	页
动态?	否
验证	无
何时更改	要测试以较少的物理内存运行系统的效果时。因为此参数不考虑核心内核和数据使用的内存以及启动进程中早期分配的各种其他数据结构使用的内存, 因此 <code>physmem</code> 的值应该小于表示较少内存量的实际页数。
交付级别	不稳定

default_stksize

说明	指定所有线程的缺省栈大小。任何线程创建时具有的栈大小都不能小于 <code>default_stksize</code> 。如果设置了 <code>default_stksize</code> , 它会覆盖 <code>lwp_default_stksize</code> 。另请参见第 27 页中的“ <code>lwp_default_stksize</code> ”。
数据类型	整数
缺省值	<ul style="list-style-type: none"> ▪ 在 SPARC 系统上为 3 x PAGESIZE ▪ 在 x64 系统上为 5 x PAGESIZE
范围	最小值为缺省值: <ul style="list-style-type: none"> ▪ 在 SPARC 系统上为 3 x PAGESIZE ▪ 在 x64 系统上为 5 x PAGESIZE

	最大值为缺省值的 32 倍。
单位	<code>getpagesize</code> 参数返回的值的倍数字节。有关更多信息，请参见 getpagesize(3C) 。
动态?	是。影响在变量更改后创建的线程。
验证	必须大于或等于 8192 且小于或等于 262,144 (256 x 1024)。还必须是系统页大小的倍数。如果不满足这些条件，将显示以下消息： <code>Illegal stack size, Using N</code> <code>N</code> 的值是 <code>default_stksize</code> 的缺省值。
何时更改	当系统因为用完栈空间而崩溃时。此问题的最佳解决方法是确定系统为何用完空间，然后进行纠正。 增大缺省栈大小意味着几乎每个内核线程都将具有更大的栈，从而导致内核内存消耗增加却没有合理原因。一般情况下，该空间将处于未使用状态。消耗增加意味着争用同一内存池的其他资源的可用空间将减少，从而可能降低系统执行工作的能力。其中一个负面影响是内核可以创建的线程数将减少。此解决方法只能作为在纠正根本原因之前采取的临时解决方法。
交付级别	不稳定

`lwp_default_stksize`

说明	指定在创建内核线程时，当调用例程没有提供要使用的明确大小时，要使用的栈大小的缺省值。
数据类型	整数
缺省值	<ul style="list-style-type: none"> ▪ 对于 SPARC 平台为 32,768 ▪ 对于 x64 平台为 20,480
范围	<p>最小值为缺省值：</p> <ul style="list-style-type: none"> ▪ 在 SPARC 系统上为 3 x <code>PAGESIZE</code> ▪ 在 x64 系统上为 5 x <code>PAGESIZE</code> <p>最大值为缺省值的 32 倍。</p>
单位	<code>getpagesize</code> 参数返回的值的倍数字节。有关更多信息，请参见 getpagesize(3C) 。
动态?	是。影响在变量更改后创建的线程。

验证	必须大于或等于 8192 且小于或等于 262,144 (256 x 1024)。还必须是系统页大小的倍数。如果不满足这些条件，将显示以下消息： Illegal stack size, Using N N 的值是 <code>lwp_default_stksize</code> 的缺省值。
何时更改	当系统因为用完栈空间而崩溃时。此问题的最佳解决方法是确定系统为何用完空间，然后进行纠正。 增大缺省栈大小意味着几乎每个内核线程都将具有更大的栈，从而导致内核内存消耗增加却没有合理原因。一般情况下，该空间将处于未使用状态。消耗增加意味着争用同一内存池的其他资源的可用空间将减少，从而可能降低系统执行工作的能力。其中一个负面影响是内核可以创建的线程数将减少。此解决方法只能作为在纠正根本原因之前采取的临时解决方法。
交付级别	不稳定

logevent_max_q_sz

说明	允许排队并等待传送到 <code>syseventd</code> 守护进程的最大系统事件数。一旦系统事件队列的大小达到此限制，队列中将不允许任何额外的系统事件。
数据类型	整数
缺省值	5000
范围	0 - MAXINT
单位	系统事件
动态?	是
验证	每次 <code>ddi_log_sysevent</code> 和 <code>sysevent_post_event</code> 生成系统事件时，系统事件框架都会检查该值。 有关更多信息，请参见 <code>ddi_log_sysevent(9F)</code> 和 <code>sysevent_post_event(3SYSEVENT)</code> 。
何时更改	当错误日志消息指出无法记录、生成或发布系统事件时。
交付级别	不稳定

segkpsize

说明	指定可用的内核可分页内存量。此内存主要用于内核线程栈。增大该值将允许为相同数量的线程或更多线程使用更大的栈。
----	--

数据类型	无符号长整数
缺省值	nCPUs/128 或物理内存量/256 GB 中的较小者 x 2 GB
范围	512 MB - 64 GB (SPARC) 200 MB - 8 GB (x86)
单位	页
动态?	否
验证	值将与最小大小和最大大小进行比较。如果小于最小值或大于最大值，会将该值重置为 2 GB。将显示一条消息来指示该结果。 在 SPARC 系统上， <code>segkpsize</code> 值不能超过物理内存大小的两倍。在 x86 系统上，该值不能超过物理内存的大小。
何时更改	需要更改此参数以在系统上支持大量进程时。缺省大小 (2 GB) 允许为超过 65,536 个内核线程创建 24 KB 的栈。如果需要大于该值或需要增大栈大小，可以增大 <code>segkpsize</code> ，前提是有足够的物理内存。
交付级别	不稳定
变更历史记录	有关信息，请参见第 158 页中的“ <code>segkpsize</code> ”。

noexec_user_stack

说明	将栈标记为不可执行的，这有助于更好地阻止缓冲区溢出攻击。 运行有 64 位内核的 Oracle Solaris 系统在缺省情况下使所有 64 位应用程序的栈成为不可执行的。要使 32 位应用程序成为不可执行的，需要设置此参数。
数据类型	带符号整数
缺省值	0 (禁用)
范围	0 (禁用) 或 1 (启用)
单位	切换 (开/关)
动态?	是。不影响当前正在运行的进程，仅影响在设置该值后创建的进程。
验证	无
何时更改	应始终启用，除非应用程序故意将可执行代码放在栈上，且没有使用 <code>mprotect</code> 使栈成为可执行的。有关更多信息，请参见 <code>mprotect(2)</code> 。
交付级别	不稳定

fsflush 和相关参数

本节介绍了 fsflush 和相关的可调参数。

fsflush

系统守护进程 fsflush 会定期运行来执行三个主要任务：

1. 每次调用时，fsflush 都会将存在时间超过指定期限的脏文件系统页刷新到磁盘。
2. 每次调用时，fsflush 都会检查内存的一部分，并促使已修改的页写入它们的后备存储。如果页已修改并且不符合以下某个条件，则会被写入：
 - 页是内核页
 - 页是空闲的
 - 页被锁定
 - 页与某个交换设备相关联
 - 页当前参与了某个 I/O 操作

实际结果是将刷新文件中通过 mmap 以写权限映射的页以及已实际更改的页。

页将刷新到后备存储中，但是保持附加到使用它们的进程。这样，如果页自刷新以来未进行修改，可以通过避免在声明页之前延迟将页写入后备存储，因而当系统在内存不足情况下运行时，这将可以简化页回收。

3. fsflush 将文件系统元数据写入到磁盘。写入是在每个第 n 次调用时执行的，其中 n 是根据各种配置变量计算得出的。有关详细信息，请参见第 30 页中的“tune_t_fsflushr”和第 31 页中的“autoup”。

以下功能是可配置的：

- 调用频率 (tune_t_fsflushr)
- 是否执行内存扫描 (dopageflush)
- 是否进行文件系统数据刷新 (doiflush)
- 进行文件系统数据刷新的频率 (autoup)

对于大多数系统，内存扫描和文件系统元数据同步是 fsflush 的主要活动。根据系统使用情况，内存扫描可能只使用很少 CPU 时间，也可能会占用太多的 CPU 时间。

tune_t_fsflushr

说明	指定 fsflush 调用之间的秒数
数据类型	带符号整数
缺省值	1

范围	1 - MAXINT
单位	秒
动态?	否
验证	如果该值小于或等于零，则该值将重置为 1 并显示一条警告消息。此检查仅在引导时执行。
何时更改	请参见 autoup 参数。
交付级别	不稳定

autoup

说明	与 <code>tune_t_flushr</code> 配合使用时， <code>autoup</code> 控制每次调用中查找脏页时要检查的内存量以及文件系统同步操作的频率。 <code>autoup</code> 的值还用于控制是否从空闲表写出缓冲区。如果缓冲区标有 <code>B_DELWRI</code> 标志（它标识已更改的文件内容页），只要该缓冲区位于列表中的时间已超过 <code>autoup</code> 秒，就将写出该缓冲区。增大 <code>autoup</code> 的值会将缓冲区在内存中保留更长时间。
数据类型	带符号整数
缺省值	30
范围	1 - MAXINT
单位	秒
动态?	否
验证	如果 <code>autoup</code> 小于或等于零，则会将其重置为 30 并显示一条警告消息。此检查仅在引导时执行。
隐式	<code>autoup</code> 应该是 <code>tune_t_fsflushr</code> 的整数倍。 <code>autoup</code> 应该至少是 <code>tune_t_fsflushr</code> 的值的 6 倍。如果不是这样，每次调用 <code>fsflush</code> 时都会扫描大量内存。 要使得在 <code>dopageflush</code> 为非零值时对内存进行检查，总系统页数乘以 <code>tune_t_fsflushr</code> 应该大于或等于 <code>autoup</code> 。
何时更改	下面是更改 <code>autoup</code> 、 <code>tune_t_fsflushr</code> 或这两者的几种可能情况： <ul style="list-style-type: none"> ■ 系统具有大量内存—在这种情况下，增大 <code>autoup</code> 可以减少每次调用 <code>fsflush</code> 时扫描的内存量。 ■ 系统具有最低内存需求—同时增大 <code>autoup</code> 和 <code>tune_t_fsflushr</code> 可以减少执行的扫描数。要保持 <code>autoup / tune_t_fsflushr</code> 的当前比率，也应增大 <code>autoup</code>。

- 具有大量瞬态文件的系统（例如，邮件服务器或软件生成计算机）— 如果创建了大量文件，然后将其删除，fsflush 可能会将这些文件的数据页写入磁盘，但这是不必要的。

交付级别 不稳定

dopageflush

说明 控制在 fsflush 调用期间是否检查内存以查找已修改的页。每次调用 fsflush 时，都将确定系统中的物理内存页数。该数值可能已经因为某个动态重新配置操作而改变。每次调用在扫描时将使用以下算法：总页数 \times `tune_t_fsflushr / autoup` 页数

数据类型 带符号整数

缺省值 1（启用）

范围 0（禁用）或 1（启用）

单位 切换（开/关）

动态? 是

验证 无

何时更改 如果系统页扫描程序极少运行（由 `vmstat` 输出的 `sr` 列中的 0 值予以指示）。

交付级别 不稳定

doiflush

说明 控制在 fsflush 调用期间是否要执行文件系统元数据同步。此同步是在 fsflush 的每个第 N 次调用时执行的，其中 $N = (\text{autoup} / \text{tune_t_fsflushr})$ 。因为此算法是整数除法，代码会进行检查以查看其迭代计数器的值是否大于或等于 N ，所以如果 `tune_t_fsflushr` 大于 `autoup`，则每次调用 `sf ush` 时都将执行同步。请注意，只有在调用 fsflush 时对 N 计算一次。随后对 `tune_t_fsflushr` 或 `autoup` 的更改对同步操作的频率没有影响。

数据类型 带符号整数

缺省值 1（启用）

范围 0（禁用）或 1（启用）

单位 切换（开/关）

动态?	是
验证	无
何时更改	一段期间内频繁修改文件且刷新导致的负载干扰了系统行为时。
	<p>如果系统重新引导能够更好地保留在某个 TMPFS 文件系统（例如 /tmp）中，则那些文件的存在及其状态一致性也不再会造成问题。通过使用 <code>mount -noatime</code> 选项，可以减少系统上的 inode 通信。此选项使得在访问文件时不进行 inode 更新。</p> <p>对于参与实时处理的系统，您可能希望禁用此选项并使用显式应用程序文件同步来获得一致性。</p>
交付级别	不稳定

进程大小调整参数

有多个参数（或变量）用于控制系统上的可用进程数以及单个用户可以创建的进程数。基础参数是 `maxusers`。该参数派生了指定给 `max_nprocs` 和 `maxuprc` 的值。

maxusers

说明	<p>最初，<code>maxusers</code> 定义了系统可以支持的登录用户数。当生成内核时，会基于该设置确定各个表的大小。当前 Oracle Solaris 发行版基于系统上的内存量执行大多数大小调整工作。因此，<code>maxusers</code> 过去的许多用途已经发生变化。依旧由 <code>maxusers</code> 派生的一些子系统包括：</p> <ul style="list-style-type: none"> ■ 系统上的最大进程数 ■ 系统中的配额结构数 ■ 目录名称查找高速缓存 (directory name look-up cache, DNLC) 的大小
数据类型	带符号整数
缺省值	以 MB 为单位的内存量与 2048 中的较小者，上述较小值与 <code>nCPUs x 8</code> 中的较大者
范围	<p>下限为 1，上限为 2048 或 <code>nCPUs x 8</code> 中的较大者（如果未在 <code>/etc/system</code> 文件中设置，上限基于物理内存大小）</p> <p>下限为 1，上限为 4096 或 <code>nCPUs x 8</code> 中的较大者（如果在 <code>/etc/system</code> 文件中做了设置）</p>
单位	用户

动态?	否。计算具有依赖性的参数后，不会再次引用 <code>maxusers</code> 。
验证	如果值大于允许的最大值，会将该值重置为最大值。将显示一条消息来指示该结果。
何时更改	系统派生的缺省用户进程数过低时。系统控制台上显示以下消息时表明发生了这种情况： out of processes 您还可能在缺省进程数过高时更改此参数，如以下情况中所示： <ul style="list-style-type: none"> ■ 当减小了 <code>maxusers</code> 的缺省值时，如果数据库服务器具有大量内存和相对较少的正在运行的进程，则可以节省系统内存。 ■ 如果文件服务器具有大量内存和较少的正在运行的进程，您可以减小该值。不过，您应该显式设置 DNLC 的大小。请参见第 57 页中的“<code>ncsize</code>”。
交付级别	不稳定
变更历史记录	有关信息，请参见第 158 页中的“ <code>maxusers</code> ”。

reserved_procs

说明	指定在进程表中要为 UID 为 <code>root(0)</code> 的进程保留的系统进程槽数。例如， <code>fsflush</code> 的 UID 为 <code>root(0)</code> 。
数据类型	带符号整数
缺省值	5
范围	5 - MAXINT
单位	进程
动态?	否。在初次参数计算后将不再使用。
验证	将与任何 <code>/etc/system</code> 设置进行匹配。
交付级别	不稳定
何时更改	考虑增大到 10+ 系统上的正常 UID 0 (<code>root</code>) 进程数时。当系统没有其他办法创建用户级进程而需要获取 <code>root shell</code> 时，该设置可以提供一定的缓冲。

pidmax

说明	指定最大可能进程 ID 的值。
----	-----------------

	<p>pidmax 设置 maxpid 变量的值。一旦设置了 maxpid，将会忽略 pidmax。maxpid 在内核的其他位置中用来确定最大进程 ID 并用于验证检查。</p> <p>通过在 /etc/system 文件中添加条目来设置 maxpid 的任何尝试都没有效果。</p>
数据类型	带符号整数
缺省值	3,000
范围	5 - 999,999
单位	进程
动态?	否。仅在引导时用来设置 pidmax 的值。
验证	是。该值将与 reserved_procs 的值和 999,999 进行比较。如果该值小于 reserved_procs 或大于 999,999，则该值将被设置为 999,999。
隐式	max_nprocs 范围检查将确保 max_nprocs 始终小于或等于该值。
何时更改	需要更改此参数以便在系统上支持 30,000 个以上的进程时。另请参见第 35 页中的“max_nprocs”。
交付级别	不稳定
变更历史记录	有关信息，请参见第 158 页中的“pidmax”。

max_nprocs

说明	<p>指定可以在系统上创建的最大进程数。包括系统进程和用户进程。在计算 maxuprc 时将使用 /etc/system 中指定的任何值。</p> <p>在确定多个其他系统数据结构的大小时也将使用该值。需要用到此参数的其他数据结构如下所述：</p> <ul style="list-style-type: none"> ■ 确定目录名称查找高速缓存的大小（如果未指定 ncsiz） ■ 验证所配置的系统 V 信号使用的内存量没有超出系统限制 ■ 为 x86 平台配置硬件地址转换资源
数据类型	带符号整数
缺省值	<p>10 + (16 x maxusers) 如果在 /etc/system 文件中设置了 maxusers</p> <p>30,000 或 10 + (128 x CPU 数目) 中的较大者，如果未在 /etc/system 文件中设置 maxusers</p>
范围	26 - maxpid 的值
动态?	否

验证	是。如果值超过 <code>maxpid</code> ，该值将设置为 <code>maxpid</code> 。
何时更改	更改此参数是在系统上支持 30,000 个以上的进程所需的步骤之一。
交付级别	不稳定
变更历史记录	有关信息，请参见第 158 页中的“ <code>max_nprocs</code> ”。

maxuprc

说明	指定任何一个用户可以在系统上创建的最大进程数。
数据类型	带符号整数
缺省值	<code>max_nprocs - reserved_procs</code>
范围	<code>1 - (max_nprocs - reserved_procs)</code>
单位	进程
动态?	否
验证	是。该值将与 <code>(max_nprocs - reserved_procs)</code> 进行比较并设置为这两个值中的较小值。
何时更改	当您希望为用户可以创建的进程数指定的硬限制小于系统可以创建的缺省进程数时。任何超出该限制的尝试将在控制台上或在消息文件中生成以下警告消息： <code>out of per-user processes for uid N</code>
交付级别	不稳定

ngroups_max

说明	指定每进程的最大补充组数。
数据类型	带符号整数
缺省值	16
范围	0 - 1024
单位	组
动态?	否
验证	是。如果 <code>ngroups_max</code> 设置为无效值，则该值将自动重置为最接近的合法值。例如，如果该值设置为小于零的值，它将重置为 0。如果该值设置为大于 1024 的值，它将重置为 1024。

何时更改	如果您在使用 NFS AUTH_SYS 验证并想要增大 ngroups_max 缺省值，请注意以下事项： <ol style="list-style-type: none"> 1. 如果 ngroups_max 设置为 16 或者如果提供的客户机的 AUTH_SYS 凭证小于等于 15 个组，将使用客户机的组信息。 2. 如果 ngroups_max 设置为大于 16 并且名称服务器中客户机的 AUTH_SYS 凭证正好包含 16 个组，则允许最大值，NFS 服务器将访问名称服务器并将客户机的 UID 与用户名相匹配。然后，名称服务器计算用户所属的组列表。
交付级别	不稳定
变更历史记录	有关信息，请参见第 158 页中的“ngroups_max”。

分页相关参数

Solaris OS 使用需求换页式虚拟内存系统。系统运行时，会根据需要将页引入内存中。当内存占用达到特定阈值之上并且对内存的需求仍在继续，将开始换页。换页要经过由特定参数控制的多个级别。

常规换页算法如下所述：

- 检测到内存不足。页扫描程序线程运行并开始遍历内存。使用包含两个步骤的算法：
 1. 将页标记为未使用的。
 2. 如果在经过一个时间间隔后仍处于未使用状态，该页将被视为要回收的主体。

如果页已经修改，则会向页面调出线程发出请求以针对该页面调度 I/O。同时，页扫描程序继续查看内存。页面调出将促使该页被写入其后备存储并置于空闲表中。页扫描程序扫描内存时，不会对页的来源进行区分。页可能来自数据文件，也可能是来自可执行文件的文本、数据或栈的页。

- 随着系统上的内存压力增加，算法在考虑备选回收页方面以及在运行换页算法的频率方面将变得更为主动。（有关更多信息，请参见第 43 页中的“fastscan”和第 43 页中的“slowscan”。）因为可用内存处于范围 lotsfree 和 minfree 之间，所以系统将线性增加每次调用页面调出线程时扫描的内存量，从由 slowscan 指定的值到由 fastscan 指定的值。系统使用 desfree 参数控制关于资源使用和行为的决策数。

系统最初限制自身使用不超过一个 CPU 的 4% 来进行 pageout 操作。随着内存压力增加，支持 pageout 操作时消耗的 CPU 时间量将线性增加，直到最大值，即一个 CPU 的 80%。算法将检查一定量的内存（介于 slowscan 和 fastscan 之间），然后在发生下列情况之一时停止：

- 已经找到了足够的页来解决内存不足问题。
- 已经查看了计划的页数。
- 已经消耗了过多时间。

当页面调出线程完成其扫描时，如果仍存在内存不足问题，则会调度在将来执行 1/4 秒的另一扫描。

换页子系统的配置机制已更改。不是依赖于 `fastscan`、`slowscan` 和 `handspreadpages` 的一组预定义值，系统将在引导时为这些参数确定适当的设置。在 `/etc/system` 文件中设置这些参数中的任何一个都会导致系统不能使用最佳值。



注意 - 请从 `/etc/system` 文件中删除 VM 系统的所有调优设置。请以缺省设置运行并确定是否需要调整这些参数中的任何参数。不要设置 `cachefree` 或 `priority_paging`。

支持对 CPU 和内存的动态重新配置 (Dynamic reconfiguration, DR)。涉及内存添加或删除的 DR 操作中的系统会重新计算相关参数的值，除非已在 `/etc/system` 中显式设置了这些参数。如果已显式设置，将使用 `/etc/system` 中指定的值，除非违反了对变量值的约束。如果那样，将重置该值。

lotsfree

说明	充当启动系统换页操作的初始触发器。当超过此阈值时，页扫描程序将被激活以开始查找要回收的内存页。
数据类型	无符号长整数
缺省值	物理内存的 1/64 和 512 KB 中的较大者
范围	最小值是 512 KB 和物理内存的 1/64 中的较大者，表示为页数（使用 <code>getpagesize</code> 返回的页大小）。有关更多信息，请参见 getpagesize(3C) 。 最大值是物理内存页数。最大值应该不超过物理内存的 30%。系统不会强制实施此范围，但“验证”部分所述内容除外。
单位	页
动态?	是，但如果发生基于内存的 DR 操作，则动态更改将丢失。
验证	如果 <code>lotsfree</code> 大于物理内存量，则其值将被重置为缺省值。
隐式	<code>lotsfree</code> 大于 <code>desfree</code> ，后者又大于 <code>minfree</code> ，应始终保持这种关系。
何时更改	如果页需求突然急剧增加，内存算法可能无法跟上需求。一个解决方法是提早开始回收内存。此解决方法可以为换页系统提供一些额外余量。 一个经验法则是将此参数设置为系统在数秒内需要分配的数量的 2 倍。此参数受工作负荷影响。DBMS 服务器在缺省设置下可能工作良好。不过，对于执行繁重的文件系统 I/O 的系统，您可能需要调整此参数。

对于具有相对静态的工作负荷和大量内存的系统，请降低此值。最小可接受值为 512 KB，表示为页数（使用 `getpagesize` 返回的页大小）。

交付级别 不稳定

desfree

说明 指定在系统上始终空闲的合适内存量。

数据类型 无符号整数

缺省值 `lotsfree / 2`

范围 最小值是 256 KB 和物理内存的 1/128 中的较大者，表示为页数（使用 `getpagesize` 返回的页大小）。

最大值是物理内存页数。最大值应该不超过物理内存的 15%。系统不会强制实施此范围，但“验证”部分所述内容除外。

单位 页

动态? 是，除非发生添加或删除内存的动态重新配置操作。在那种情况下，该值将重置为 `/etc/system` 文件中提供的值或者将根据新的物理内存值计算得出。

验证 如果 `desfree` 大于 `lotsfree`，则 `desfree` 将设置为 `lotsfree / 2`。不会显示任何消息。

隐式 `lotsfree` 大于 `desfree`，后者又大于 `minfree`，应始终保持这种关系。

负面影响 增大此参数的值可能会带来多种负面影响。当新值接近或超过系统上的可用内存量时，会发生以下情况：

- 将不处理异步 I/O 请求，除非可用内存超过 `desfree`。增大 `desfree` 的值可能会导致本应可以成功的请求被拒绝。
- NFS 异步写入被作为同步写入执行。
- 交换程序将提前激活，交换程序的行为将倾向于更主动的操作。
- 系统可能无法将尽可能多的可执行页预装入系统。此负面影响可能会导致应用程序的运行速度比它们应有的速度要慢。

何时更改 对于具有相对静态的工作负荷和大量内存的系统，请降低此值。最小可接受值为 256 KB，表示为页数（使用 `getpagesize` 返回的页大小）。

交付级别 不稳定

minfree

说明	指定最小可接受内存级别。当内存低于此数值时，系统将优先安排成功完成页面调出操作或者将进程完全交换出内存所需的分配。分配将拒绝或者阻塞其他分配请求。
数据类型	无符号整数
缺省值	<code>desfree / 2</code>
范围	最小值是 128 KB 和物理内存的 1/256 中的较大者，表示为页数（使用 <code>getpagesize</code> 返回的页大小）。 最大值是物理内存页数。最大值应该不超过物理内存的 7.5%。系统不会强制实施此范围，但“验证”部分所述内容除外。
单位	页
动态?	是，除非发生添加或删除内存的动态重新配置操作。在那种情况下，该值将重置为 <code>/etc/system</code> 文件中提供的值或者将根据新的物理内存值计算得出。
验证	如果 <code>minfree</code> 大于 <code>desfree</code> ，则 <code>minfree</code> 将设置为 <code>desfree / 2</code> 。不会显示任何消息。
隐式	<code>lotsfree</code> 大于 <code>desfree</code> ，后者又大于 <code>minfree</code> ，应始终保持这种关系。
何时更改	缺省值通常就足够了。对于具有相对静态的工作负荷和大量内存的系统，请降低此值。最小可接受值为 128 KB，表示为页数（使用 <code>getpagesize</code> 返回的页大小）。
交付级别	不稳定

throttlefree

说明	指定内存级别，在该级别，阻塞的内存分配请求将被置于休眠状态，即使内存足以满足该请求。
数据类型	无符号整数
缺省值	<code>minfree</code>
范围	最小值是 128 KB 和物理内存的 1/256 中的较大者，表示为页数（使用 <code>getpagesize</code> 返回的页大小）。 最大值是物理内存页数。最大值应该不超过物理内存的 4%。系统不会强制实施此范围，但“验证”部分所述内容除外。
单位	页

动态?	是，除非发生添加或删除内存的动态重新配置操作。在那种情况下，该值将重置为 <code>/etc/system</code> 文件中提供的值或者将根据新的物理内存值计算得出。
验证	如果 <code>throttlefree</code> 大于 <code>desfree</code> ，则 <code>throttlefree</code> 将设置为 <code>minfree</code> 。不会显示任何消息。
隐式	<code>lotsfree</code> 大于 <code>desfree</code> ，后者大于 <code>minfree</code> ，应始终保持这种关系。
何时更改	缺省值通常就足够了。对于具有相对静态的工作负荷和大量内存的系统，请降低此值。最小可接受值为 128 KB，表示为页数（使用 <code>getpagesize</code> 返回的页大小）。有关更多信息，请参见 getpagesize(3C) 。
交付级别	不稳定

pageout_reserve

说明	指定保留供页面调出或调度程序线程专用的页数。当可用内存小于该值时，对于除页面调出或调度程序之外的进程，将拒绝非阻塞分配。页面调出线程需要有一个小内存池供其使用，以便它可以分配执行 I/O 来将页写入其后备存储时所需的数据结构。
数据类型	无符号整数
缺省值	<code>throttlefree / 2</code>
范围	最小值是 64 KB 和物理内存的 1/512 中的较大者，表示为页数（使用 <code>getpagesize(3C)</code> 返回的页大小）。 最大值是物理内存页数。最大值应该不超过物理内存的 2%。系统不会强制实施此范围，但“验证”部分所述内容除外。
单位	页
动态?	是，除非发生添加或删除内存的动态重新配置操作。在那种情况下，该值将重置为 <code>/etc/system</code> 文件中提供的值或者将根据新的物理内存值计算得出。
验证	如果 <code>pageout_reserve</code> 大于 <code>throttlefree / 2</code> ，则 <code>pageout_reserve</code> 将设置为 <code>throttlefree / 2</code> 。不会显示任何消息。
隐式	<code>lotsfree</code> 大于 <code>desfree</code> ，后者又大于 <code>minfree</code> ，应始终保持这种关系。
何时更改	缺省值通常就足够了。对于具有相对静态的工作负荷和大量内存的系统，请降低此值。最小可接受值为 64 KB，表示为页数（使用 <code>getpagesize</code> 返回的页大小）。
交付级别	不稳定

pages_pp_maximum

说明	定义必须处于未锁定状态的页数。如果将锁定页的某个请求会迫使可用内存低于该值，将会拒绝该请求。
数据类型	无符号长整数
缺省值	(<code>tune_t_minarmem + 100</code> 和 [引导时可用内存的 4% + 4 MB]) 中的较大者
范围	系统强制实施的最小值是 <code>tune_t_minarmem + 100</code> 。系统不会强制实施最大值。
单位	页
动态?	是，除非发生添加或删除内存的动态重新配置操作。在那种情况下，该值将重置为 <code>/etc/system</code> 文件中提供的值或者将根据新的物理内存值计算得出。
验证	如果 <code>/etc/system</code> 文件中指定的值或计算得出的缺省值小于 <code>tune_t_minarmem + 100</code> ，则该值将重置为 <code>tune_t_minarmem + 100</code> 。 如果增大 <code>/etc/system</code> 文件中的该值，则不会显示任何消息。仅在引导时，以及在涉及添加或删除内存的动态重新配置操作期间，才会进行验证。
何时更改	当可用内存量看起来足够用，但内存锁定请求却失败或者向共享内存段附加 <code>SHARE_MMU</code> 标志失败时。 太大的值会导致内存锁定请求 (<code>mlock</code> 、 <code>mlockall</code> 和 <code>memcntl</code>) 无谓地失败。有关更多信息，请参见 <code>mlock(3C)</code> 、 <code>mlockall(3C)</code> 和 <code>memcntl(2)</code> 。
交付级别	不稳定

tune_t_minarmem

说明	定义为避免死锁而需要维持的最小可用驻留（不可交换）内存。用来保留部分内存供 OS 的核心使用。在 OS 确定最大可用内存量时，以此方式限制的页将不可见。
数据类型	带符号整数
缺省值	25
范围	1 - 物理内存
单位	页
动态?	否
验证	无。较大的值会导致物理内存浪费。

何时更改 缺省值通常就足够了。如果系统发生锁定并且调试信息指示没有内存可用，请考虑增大缺省值。

交付级别 不稳定

fastscan

说明 定义内存压力最高时系统每秒查看的最大页数。

数据类型 带符号整数

缺省值 请以下列方式之一设置 fastscan 缺省值：

- 使用 /etc/system 文件中设置的 fastscan 值。
- 使用 /etc/system 文件中设置的 maxfastscan 值。
- 如果 /etc/system 文件中既没有设置 fastscan 也没有设置 maxfastscan，fastscan 将在系统引导时设置为 64 MB。然后，在系统引导几分钟后，fastscan 值将设置为扫描程序使用一个 CPU 的 10% 在一秒内可以扫描的页数。

在以上所有三种情况中，如果派生的值大于系统中内存的一半，fastscan 值将被限定为系统中内存值的一半。

范围 64 MB - 系统物理内存的一半

单位 页

动态? 是，除非发生添加或删除内存的动态重新配置操作。在那种情况下，该值将重置为 /etc/system 文件中提供的值或者将根据新的物理内存值计算得出。

验证 最大值是 64 MB 和物理内存的 1/2 中的较小者。

何时更改 当内存不足期间首选更主动的内存扫描时，特别是当系统受制于密集内存需求时，或者当执行繁重文件 I/O 时。

交付级别 不稳定

slowscan

说明 定义尝试回收内存时系统每秒查看的最小页数。

数据类型 带符号整数

缺省值 物理内存的 1/20（以页数表示）和 100 中的较小者。

范围 $1 - \text{fastscan} / 2$

单位	页
动态?	是，除非发生添加或删除内存的动态重新配置操作。在那种情况下，该值将重置为 <code>/etc/system</code> 文件中提供的值或者将根据新的物理内存值计算得出。
验证	如果 <code>slowscan</code> 大于 <code>fastscan/2</code> ，则 <code>slowscan</code> 将重置为 <code>fastscan/2</code> 。不会显示任何消息。
何时更改	当内存不足期间首选更主动的内存扫描时，特别是当系统受制于密集内存需求时。
交付级别	不稳定

min_percent_cpu

说明	定义 <code>pageout</code> 可以使用的最小 CPU 百分比。在确定页扫描程序可以使用的最大时间量时，可以使用此参数作为起点
数据类型	带符号整数
缺省值	4
范围	1 - 80
单位	百分比
动态?	是
验证	无
何时更改	在具有多个 CPU 和大量内存的系统上，如果系统受制于密集内存需求，增大该值可以允许页面调度程序花费更多时间来尝试查找内存。
交付级别	不稳定

handspreadpages

说明	Oracle Solaris OS 使用双指针时钟算法来查找当内存较低时要回收的页。时钟的第一个指针遍历内存，将页标记为未使用的。第二个指针在第一个指针之后一段距离遍历内存，查看页是否仍标记为未使用的。如果页仍标记为未使用的，该页将被回收。第一个指针和第二个指针之间的距离为 <code>handspreadpages</code> 。
数据类型	无符号长整数
缺省值	<code>fastscan</code>
范围	1 - 系统中的最大物理内存页数

单位	页
动态?	是。此参数要求内核 <code>reset_hands</code> 参数也设置为非零值。在 <code>handsreadpages</code> 的新值被认可后， <code>reset_hands</code> 将设置为零。
验证	该值将设置为物理内存量和 <code>handsreadpages</code> 值中的较小者。
何时更改	当您要增大页在被回收之前可能驻留的时间量时。增大该值会增大两个指针之间的间隔，因此，会增大回收页之前的时间量。
交付级别	不稳定

pages_before_pager

说明	定义在 I/O 完成后立即释放页（而不是存储页以便可以重用）的部分系统阈值。阈值为 <code>lotsfree + pages_before_pager</code> 。NFS 环境还使用此阈值在内存压力增加时减少其异步活动。
数据类型	带符号整数
缺省值	200
范围	1 - 物理内存量
单位	页
动态?	否
验证	无
何时更改	如果绝大多数 I/O 是针对实际上只读取或写入了一次便未再引用的页执行的，则您可以更改此参数。将此变量设置为较大的内存量可以保持向空闲表添加页。 此外，当系统受制于突发的严重内存压力时，您也可以更改此参数。在这种情况下，较大的值有助于保持较大的缓冲来应对压力。
交付级别	不稳定

maxpgio

说明	定义换页系统可以排队的最大页 I/O 请求数。此数值除以 4 可以得到分页系统使用的最大实际数值。此参数用于限制请求数以及控制进程交换。
数据类型	带符号整数
缺省值	40

范围	1 到一个可变的最大值，该最大值取决于系统体系结构，但是主要取决于 I/O 子系统，例如控制器数、磁盘数和磁盘交换区大小
单位	I/O
动态?	否
验证	无
隐式	来自页面调度程序的最大 I/O 请求数受请求缓冲区列表大小的限制，该大小当前为 256。
何时更改	需要增大此参数以便更快地换出内存时。如果配置了多个交换设备或者如果交换设备为分散读写设备，较大的值可能有助于更快地缓解内存压力。请注意，现有 I/O 子系统应该能够处理额外 I/O 负载。此外，如果交换分区和应用程序文件位于同一磁盘上，增大交换区 I/O 可能会降低应用程序 I/O 性能。
交付级别	不稳定

与交换相关的参数

Oracle Solaris OS 中的交换操作是由 `swapfs` 伪文件系统完成的。交换设备上的空间与物理内存的组合被视为空间池，可用于为系统提供支持以便为匿名内存维护后备存储。系统首先尝试从磁盘设备分配空间，然后使用物理内存作为后备存储。当 `swapfs` 被迫将系统内存用于后备存储时，将强制实施限制以确保系统不会因为 `swapfs` 的过度使用而发生死锁。

`swapfs_reserve`

说明	定义要保留供系统 (UID = 0) 进程使用的内存量。
数据类型	无符号长整数
缺省值	4 MB 和物理内存的 1/16 中的较小者
范围	最小值是 4 MB 和物理内存的 1/16 中的较小者，表示为页数（使用 <code>getpagesize</code> 返回的页大小）。 最大值是物理内存页数。最大值应该不超过物理内存的 10%。系统不会强制实施此范围，但“验证”部分所述内容除外。
单位	页
动态?	否
验证	无

何时更改	一般情况下不需要更改。仅当软件提供商建议更改时，或者当系统进程因无法获取交换空间而终止时才应更改。一个更好的解决方法是向系统添加物理内存或者其他交换设备。
交付级别	不稳定

swapfs_minfree

说明	定义应保留为空闲状态以供系统的其余部分使用的必需物理内存量。当尝试保留内存以用作某个进程的交换空间时，如果系统感到可用内存将变得低于该值，则会拒绝这些尝试。以此方式保留的页仅能用于由内核或用户级进程执行的锁定分配。
数据类型	无符号长整数
缺省值	2 MB 和物理内存的 12.5% 中的较大者
范围	1 - 物理内存量
单位	页
动态?	否
验证	无
何时更改	当进程因为无法获取交换空间而失败，但系统仍有可用内存时，考虑减小此参数值。例如，可更改此值以使用不超过 6.25% 的系统内存，但是当低于系统内存的 5% 时不要减小该值。 在 SPARC 系统上，该值应该至少是 <code>tsb_alloc_hiwater_factor</code> 值的 2 倍。有关更多信息，请参见第 69 页中的“ <code>tsb_alloc_hiwater_factor</code> ”。
交付级别	不稳定
变更历史记录	有关信息，请参见第 159 页中的“ <code>swapfs_minfree</code> ”。

内核内存分配器

Oracle Solaris 内核内存分配器在内核中分布内存块以供客户机使用。分配器创建大小不同的许多高速缓存供其客户机使用。客户机还可以请求分配器创建高速缓存以供其自己使用（例如，分配特定大小的结构）。可以使用 `kstat -c kmem_cache` 命令查看分配器管理的每个高速缓存的相关统计信息。

有时，系统可能因为内存损坏而崩溃。内核内存分配器支持一个调试接口（一组标志），该接口对缓冲区执行各种完整性检查。内核内存分配器还收集关于分配器的信

息。通过完整性检查，可以在错误实际发生之前检测到这些错误。收集的信息在支持人员尝试查明崩溃原因时为他们提供一些额外数据。

使用这些标志会导致在系统操作过程中产生额外的系统开销和内存使用情况。仅当怀疑发生了内存损坏问题时才应使用这些标志。

kmem_flags

说明 Oracle Solaris 内核内存分配器具有各种调试和测试选项。

下面介绍了五个受支持的标志设置。

标志	设置	说明
AUDIT	0x1	分配器维护着一个日志，其中包含其最近的活动历史记录。所记录的项数取决于是否还设置了 CONTENTS。日志的大小是固定的。当空间耗尽时，将回收早期的记录。
TEST	0x2	分配器将某个模式写入已释放的内存，并在下次分配缓冲区时检查该模式是否保持未变。如果缓冲区的某些部分已更改，则先前分配并释放了该缓冲区的客户机可能已使用了内存。如果检测到覆盖，系统将会崩溃。
REDZONE	0x4	分配器在所请求的缓冲区末尾提供额外的内存并在该内存中插入一个特殊模式。当缓冲区被释放时，会检查该模式以查看写入数据时是否超出了缓冲区末尾。如果检测到覆盖，内核将会崩溃。
CONTENTS	0x8	当缓冲区被释放时，分配器记录最多 256 字节的缓冲区内容。该标志要求同时设置 AUDIT。 这些标志的数值可按逻辑方式加在一起并通过 <code>/etc/system</code> 文件进行设置。
LITE	0x100	分配和释放缓冲区时执行最小完整性检查。启用了此标志时，分配器会检查是否尚未写入到红色区域中、已释放的缓冲区是否未被再次释放以及正在释放的缓冲区是否为已分配的大小。不要将此标志与任何其他标志组合使用。

数据类型 带符号整数

缺省值 0（禁用）

范围 0（禁用）或 1 - 15 或 256 (0x100)

动态?	是。运行时所做的更改仅影响新的内核内存高速缓存。在系统初始化后，极少会创建新的高速缓存。
验证	无
何时更改	怀疑发生内存损坏时。
交付级别	不稳定

kmem_stackinfo

说明	<p>如果创建内核线程时在 <code>/etc/system</code> 中启用了 <code>kmem_stackinfo</code> 变量，将会以特定模式填充内核线程栈，而不是用零填充。在内核线程执行过程中，此内核线程栈模式将被逐步覆盖。从栈顶部开始直到找不到该模式为止，一个简单计数会提供一个高水位值，该值是内核线程使用的最大内核栈空间。该机制允许以下功能：</p> <ul style="list-style-type: none"> ■ 计算实际为系统中的当前内核线程使用的内核线程栈所占的百分比（高水位） ■ 当某个内核线程结束时，对于最后那些在终止前已使用了其大部分内核线程栈的内核线程，系统会将其记录到一个小的循环内存缓冲区中
数据类型	无符号整数
缺省值	0（禁用）
范围	0（禁用）或 1（启用）
动态?	是
验证	无
何时更改	当您希望监视内核线程栈使用情况时。请记住，当启用了 <code>kmem_stackinfo</code> 时，创建和删除 <code>kthreads</code> 时的性能会降低。有关更多信息，请参见《 Oracle Solaris Modular Debugger Guide 》中的第 5 章“ Built-In Commands ”。
区域配置	此参数必须在全局区域中设置。
交付级别	不稳定

常规驱动程序参数

moddebug

说明 当启用了此参数时，会显示关于模块装入过程中各个步骤的消息。

数据类型 带符号整数

缺省值 0（消息关闭）

范围 下面是最有用的值：

- 0x80000000—输出 [un] loading... 消息。对于装入的每个模块，将在控制台上和 /var/adm/messages 文件中显示如下所示的消息：

```
Apr 20 17:18:04 neo genunix: [ID 943528 kern.notice] load 'sched/TS_DPTBL' id 15
loaded @ 0x7be1b2f8/0x19c8380 size 176/2096
Apr 20 17:18:04 neo genunix: [ID 131579 kern.notice] installing TS_DPTBL,
module id 15.
```

- 0x40000000—输出详细的错误消息。对于装入的每个模块，将在控制台上和 /var/adm/messages 文件中显示如下所示的消息：

```
Apr 20 18:30:00 neo unix: Errno = 2
Apr 20 18:30:00 neo unix: kobj_open: vn_open of /platform/sun4v/kernel/exec/sparcv9/intpexec fails
Apr 20 18:30:00 neo unix: Errno = 2
Apr 20 18:30:00 neo unix: kobj_open: '/kernel/exec/sparcv9/intpexec'
Apr 20 18:30:00 neo unix: vp = 60015777600
Apr 20 18:30:00 neo unix: kobj_close: 0x60015777600
Apr 20 18:30:00 neo unix: kobj_open: vn_open of /platform/SUNW,Sun-Fire-T200/kernel/exec/sparcv9
/intpexec fails,
Apr 20 18:30:00 neo unix: Errno = 2
Apr 20 18:30:00 neo unix: kobj_open: vn_open of /platform/sun4v/kernel/exec/sparcv9/intpexec fails
```

- 0x20000000—输出更详细的消息。设置为此值时，在系统引导过程中，输出的信息不会比使用 0x40000000 标志时输出的信息多。不过，设置为此值时，在卸载模块时会输出与模块的释放相关的信息。

这些值可以相加在一起来设置最终值。

动态? 是

验证 无

何时更改 当模块未按预期的那样装入时，或者系统在装入模块过程中看似挂起时。请注意，当设置为 0x40000000 时，系统引导速度会相应地降低，取决于写入到控制台的消息数。

交付级别 不稳定

ddi_msix_alloc_limit

说明	仅限 x86：此参数控制设备实例可以分配的扩展消息信号中断 (Extended Message Signaled Interrupts, MSI-X) 数。由于某个现有的系统限制，缺省值为 2。通过增大此参数的值，可以增加设备实例可以分配的 MSI-X 中断数。可以在附加设备驱动程序之前通过编辑 <code>/etc/system</code> 文件或使用 <code>mdb</code> 命令来设置此参数。
数据类型	带符号整数
缺省值	2
范围	1 - 16
动态?	是
验证	无
何时更改	要增加设备实例可以分配的 MSI-X 中断数时。不过，如果您增加了设备实例可以分配的 MSI-X 中断数，不一定会有足够的中断来满足所有分配请求。如果发生这种情况，某些设备可能会停止工作或者系统可能无法引导。在这种情况下，可以减小值或者删除此参数。
交付级别	不稳定

网络驱动程序参数

igb 参数

mr_enable

说明	此参数启用或禁用 <code>igb</code> 网络驱动程序使用的多个接收和传送队列。可以在附加 <code>igb</code> 驱动程序之前通过编辑 <code>/etc/driver/drv/igb.conf</code> 文件来设置此参数。
数据类型	布尔值
缺省值	1 (禁用多个队列)
范围	0 (启用多个队列) 或 1 (禁用多个队列)
动态?	否
验证	无
何时更改	要启用或禁用 <code>igb</code> 网络驱动程序使用的多个接收和传送队列时。

交付级别 不稳定

intr_force

说明 此参数用于强制实施 `igb` 使用的中断类型，例如 MSI、MSI-X 或传统。可以在附加 `igb` 驱动程序之前通过编辑 `/etc/driver/drv/igb.conf` 文件来设置此参数。

数据类型 无符号整数

缺省值 0（不强制实施中断类型）

范围 0（不强制实施中断类型）
1（强制实施 MSI-X 中断类型）
2（强制实施 MSI 中断类型）
3（强制实施传统中断类型）

动态? 否

验证 无

何时更改 要强制实施 `igb` 网络驱动程序使用的中断类型时。

交付级别 不稳定

ixgbe 参数

tx_queue_number

说明 此参数控制 `ixgbe` 网络驱动程序使用的传送队列数。可以通过增大此参数的值来增加传送队列数。可以在附加 `ixgbe` 驱动程序之前通过编辑 `/etc/driver/drv/ixgbe.conf` 文件来设置此参数。

数据类型 无符号整数

缺省值 8

范围 1 - 32

动态? 否

验证 无

何时更改 要更改 `ixgbe` 网络驱动程序使用的传送队列数时。

交付级别 不稳定

rx_queue_number

说明	此参数控制 ixgbe 网络驱动程序使用的接收队列数。可以通过增大此参数的值来增加接收队列数。可以在附加 ixgbe 驱动程序之前通过编辑 <code>/etc/driver/drv/ixgbe.conf</code> 文件来设置此参数。
数据类型	无符号整数
缺省值	8
范围	1 - 64
动态?	否
验证	无
何时更改	要更改 ixgbe 网络驱动程序使用的接收队列数时。
交付级别	不稳定

intr_throttling

说明	此参数控制 ixgbe 网络驱动程序的中断抑制率。可以通过减小此参数的值来增大中断率。可以在附加 ixgbe 驱动程序之前通过编辑 <code>/etc/driver/drv/ixgbe.conf</code> 文件来设置此参数。
数据类型	无符号整数
缺省值	200
范围	0 至 65535
动态?	否
验证	无
何时更改	要更改 ixgbe 网络驱动程序使用的中断上限速率。
交付级别	不稳定

rx_limit_per_intr

说明	此参数控制 ixgbe 网络驱动程序使用的每中断最大接收队列缓冲区描述符数。可以通过增大此参数的值来增加接收队列缓冲区描述符数。可以在附加 ixgbe 驱动程序之前通过编辑 <code>/etc/driver/drv/ixgbe.conf</code> 文件来设置此参数。
数据类型	无符号整数
缺省值	256
范围	16 - 4096

动态?	否
验证	无
何时更改	要更改 ixgbe 网络驱动程序针对每个中断处理的接收队列缓冲区描述符数时。
交付级别	不稳定

tx_ring_size

说明	此参数控制 ixgbe 网络驱动程序使用的传送队列大小。可以通过增大此参数的值来增加传送队列大小。可以在附加 ixgbe 驱动程序之前通过编辑 /etc/driver/drv/ixgbe.conf 文件来设置此参数。
数据类型	无符号整数
缺省值	1024
范围	64 - 4096
动态?	否
验证	无
何时更改	要更改 ixgbe 网络驱动程序使用的传送队列大小时。
交付级别	不稳定

rx_ring_size

说明	此参数控制 ixgbe 网络驱动程序使用的接收队列大小。可以通过增大此参数的值来增加接收队列大小。可以在附加 ixgbe 驱动程序之前通过编辑 /etc/driver/drv/ixgbe.conf 文件来设置此参数。
数据类型	无符号整数
缺省值	1024
范围	64 - 4096
动态?	否
验证	无
何时更改	要更改 ixgbe 网络驱动程序使用的接收队列大小时。
交付级别	不稳定

tx_copy_threshold

说明	此参数控制 ixgbe 网络驱动程序使用的传送缓冲区复制阈值。可以通过增大此参数的值来增大传送缓冲区复制阈值。可以在附加 ixgbe 驱动程序之前通过编辑 <code>/etc/driver/drv/ixgbe.conf</code> 文件来设置此参数。
数据类型	无符号整数
缺省值	512
范围	0 - 9126
动态?	否
验证	无
何时更改	要更改 ixgbe 网络驱动程序使用的传送缓冲区复制阈值时。
交付级别	不稳定

rx_copy_threshold

说明	此参数控制 ixgbe 网络驱动程序使用的接收缓冲区复制阈值。可以通过增大此参数的值来增大接收缓冲区复制阈值。可以在附加 ixgbe 驱动程序之前通过编辑 <code>/etc/driver/drv/ixgbe.conf</code> 文件来设置此参数。
数据类型	无符号整数
缺省值	128
范围	0 - 9126
动态?	否
验证	无
何时更改	要更改 ixgbe 网络驱动程序使用的接收缓冲区复制阈值时。
交付级别	不稳定

常规 I/O 参数

maxphys

说明	定义物理 I/O 请求的最大大小。如果驱动程序遇到大于此大小的请求，驱动程序会将该请求分成多个大小为 <code>maxphys</code> 的块。文件系统可以并且确实施加了其自己的限制。
数据类型	带符号整数

缺省值	131,072 (sun4u 或 sun4v) 或 57,344 (x86)。如果驱动器支持宽传送, 则 sd 驱动程序使用值 1,048,576。缺省情况下, ssd 驱动程序使用 1,048,576。
范围	计算机特定的页大小 - MAXINT
单位	字节
动态?	是, 但是当挂载文件系统时, 许多文件系统会将此值装入每挂载点数据结构。将设备附加到驱动程序特定的数据结构时, 许多驱动程序会装入该值。
验证	无
何时更改	当以大块形式在原始设备之间执行 I/O 时。请注意, 执行 OLTP 操作的 DBMS 会发出大量小的 I/O。在这种情况下, 更改 maxphys 不会带来任何性能改善。
交付级别	不稳定

rlim_fd_max

说明	指定对单个进程可以打开的文件描述符数目的“硬”限制。覆盖此限制需要具有超级用户特权。
数据类型	带符号整数
缺省值	65,536
范围	1 - MAXINT
单位	文件描述符
动态?	否
验证	无
何时更改	<p>进程的最大已打开文件数不够用时。因为系统体系中还存在其他限制, 所以较大的文件描述符数目也不一定能发挥它应有的作用。例如:</p> <ul style="list-style-type: none"> 使用标准 I/O 的 32 位程序被限定为最多使用 256 个文件描述符。使用标准 I/O 的 64 位程序最多可以使用 20 亿个描述符。具体来说, 标准 I/O 是指 libc(3LIB) 中的 stdio(3C) 函数。 缺省情况下, select 被限制为每个 fd_set 可有 1024 个描述符。有关更多信息, 请参见 select(3C)。32 位应用程序代码可以使用较大的 fd_set 大小 (小于或等于 65,536) 进行重新编译。64 位应用程序使用的 fd_set 大小为 65,536, 该大小不能更改。

在系统范围内更改此值的一种替代方法是使用 `plimit(1)` 命令。如果父进程通过 `plimit` 更改了其限制，所有子进程都会继承增大的限制。对于 `inetd` 等守护进程，此替代方法非常有用。

交付级别 不稳定

rlim_fd_cur

说明	定义对单个进程可以打开的文件描述符数目的“软”限制。进程可以通过使用 <code>setrlimit()</code> 调用或者通过在它运行的任何 shell 中发出 <code>limit</code> 命令，将其文件描述符限制调整为任何值，但不能超过由 <code>rlim_fd_max</code> 定义的“硬”限制。将该限制调整为小于或等于硬限制的任何值时，不需要超级用户特权。
数据类型	带符号整数
缺省值	256
范围	1 - MAXINT
单位	文件描述符
动态?	否
验证	与 <code>rlim_fd_max</code> 进行比较。如果 <code>rlim_fd_cur</code> 大于 <code>rlim_fd_max</code> ，则 <code>rlim_fd_cur</code> 将重置为 <code>rlim_fd_max</code> 。
何时更改	进程的缺省已打开文件数不够用时。增大此值仅意味着程序可能不需要使用 <code>setrlimit</code> 来增大其可用的最大文件描述符数。
交付级别	不稳定

常规文件系统参数

ncsize

说明	定义目录名称查找高速缓存 (directory name look-up cache, DNLC) 中的条目数。UFS、NFS 和 ZFS 在缓存已解析的路径名的元素时会使用此参数。 DNLC 还缓存负查找信息，这意味着它会在高速缓存中缓存未找到的名称。
数据类型	带符号整数
缺省值	$(4 \times (v.v_proc + \text{maxusers}) + 320) + (4 \times (v.v_proc + \text{maxusers}) + 320) / 100$

范围	0 - MAXINT
单位	DNLC 条目
动态?	否
验证	无。较大的值会导致卸载文件系统时花费的时间增加，因为在卸载过程中高速缓存必须清除该文件系统的条目。
何时更改	可以使用 <code>kstat -n dnlcstats</code> 命令来确定何时因为 DNLC 太小而从中删除了条目。 <code>pick_heuristic</code> 和 <code>pick_last</code> 参数的总和表示本应当有效但因为高速缓存太小而被回收的条目数。 <code>ncsize</code> 的值过大对系统会有直接影响，因为系统将基于 <code>ncsize</code> 的值为 DNLC 分配一组数据结构。缺省情况下，系统为 <code>ncsize</code> 分配 64 字节的结构。再进一步，该值对 UFS 和 NFS 也具有影响，除非显式设置了 <code>ufs_ninode</code> 和 <code>nfs:nrnode</code> 。
交付级别	不稳定

dnlc_dir_enable

说明 启用大型目录高速缓存

注 - 此参数对 NFS 或 ZFS 文件系统没有影响。

数据类型	无符号整数
缺省值	1 (启用)
范围	0 (禁用) 或 1 (启用)
动态?	是，但不要动态更改此可调参数。如果最初禁用了此参数，您可以启用此参数。或者，如果最初启用了此参数，您可以将其禁用。但是，启用、禁用，以及之后再启用此参数可能会导致过时的目录高速缓存。
验证	否
何时更改	目录高速缓存没有已知问题。不过，如果出现问题，请将 <code>dnlc_dir_enable</code> 设置为 0，以禁用高速缓存。
交付级别	不稳定

dnlc_dir_min_size

说明 指定为一个目录缓存的最小条目数。

注 - 此参数对 NFS 或 ZFS 文件系统没有影响。

数据类型	无符号整数
缺省值	40
范围	0 - MAXUINT（无最大值）
单位	条目
动态?	是，可以随时更改此参数。
验证	无
何时更改	如果缓存小目录时出现性能问题，请增大 <code>dnlc_dir_min_size</code> 。请注意，对于目录的缓存，各个文件系统可能具有各自的范围限制。
交付级别	不稳定

dnlc_dir_max_size

说明 指定为一个目录缓存的最大条目数。

注 - 此参数对 NFS 或 ZFS 文件系统没有影响。

数据类型	无符号整数
缺省值	MAXUINT（无最大值）
范围	0 - MAXUINT
动态?	是，可以随时更改此参数。
验证	无
何时更改	如果使用大型目录出现性能问题，请减小 <code>dnlc_dir_max_size</code> 。
交付级别	不稳定

dnlc_dircache_percent

说明	计算 DNLC 目录高速缓存可以使用的最大物理内存百分比。
数据类型	整数
缺省值	100

范围	0 - 100
单位	百分比
动态?	否
验证	引导时，将检查值的范围并执行缺省值。
何时更改	系统遇到内存不足和高内核内存消耗时，可考虑减小该值。如果使用缺省值出现性能问题，可考虑增大该值。

注 - UFS 和 ZFS 文件系统以及 NFS 客户机使用 DNLC。当存在内存不足和高内核内存消耗时，或者当 ARC 或其他内核高速缓存需要内存时，可以考虑设置该可调参数，以获得更好的性能。

交付级别	不稳定
变更历史记录	有关信息，请参见第 159 页中的“dnlc_dircache_percent”。

TMPFS 参数

tmpfs:tmpfs_maxkmem

说明	定义 TMPFS 可以用于其数据结构（tmpnodes 和目录条目）的最大内核内存量。
数据类型	无符号长整数
缺省值	一页和物理内存的 4% 中的较大者。
范围	一页中的字节数（对于 sun4u 或 sun4v 系统为 8192，对于所有其他系统为 4096）至首次使用 TMPFS 时的可用内核内存的 25%。
单位	字节
动态?	是
验证	无
何时更改	如果以下消息显示在了控制台上或者写入到了消息文件中，请增大此参数： <code>tmp_memalloc: tmpfs over memory limit</code> TMPFS 用于其数据结构的当前内存量存放在 tmp_kmemspace 字段中。可以通过内核调试器检查此字段。

交付级别 不稳定

tmpfs:tmpfs_minfree

说明	定义 TMPFS 为系统其余部分保留的最小交换空间量。
数据类型	带符号长整数
缺省值	512
范围	0 - 最大交换空间大小
单位	页
动态?	是
验证	无
何时更改	要在大量使用 TMPFS 的系统上维护合理的交换空间量，您可以增大此数值。如果控制台或消息文件中显示了以下消息，则表明已经达到了限制： <i>fs-name: File system full, swap space limit exceeded</i>
交付级别	不稳定

伪终端

伪终端 `pty` 在 Oracle Solaris 软件中用于两个目的：

- 通过使用 `telnet`、`rlogin` 或 `rsh` 命令来支持远程登录
- 提供 X 窗口系统用以创建命令解释程序窗口的接口

缺省伪终端数足够供桌面工作站使用。所以，调优将着重于供远程登录使用的 `pty` 数。

缺省 `pty` 数现在基于系统上的内存量。仅当要限制或增大可以登录到系统中的用户数时才应更改此缺省值。

在配置过程中要使用三个相关变量：

- `pt_cnt`—缺省的最大 `pty` 数目。
- `pt_pctofmem`—可专用于 `pty` 支持结构的内核内存所占百分比。值为零意味着没有远程用户可以登录到系统中。
- `pt_max_pty`—硬性的 `pty` 数目最大值。

pt_cnt 的缺省值为零，这指示系统基于 pt_pctofmem 中指定的内存量限制登录数，除非设置了 pt_max_pty。如果 pt_cnt 不为零，分配的 pty 数目可以一直达到此限制。当超出该阈值时，系统将查看 pt_max_pty。如果 pt_max_pty 具有非零值，则会将其与 pt_cnt 进行比较。如果 pt_cnt 小于 pt_max_pty，将允许 pty 分配。如果 pt_max_pty 为零，pt_cnt 将与基于 pt_pctofmem 支持的 pty 数目进行比较。如果 pt_cnt 小于该值，将允许 pty 分配。请注意，仅当 pt_cnt 和 ptms_ptymax 都具有缺省值零时，基于 pt_pctofmem 的限制才开始起作用。

要对 pty 施加与从 pt_pctofmem 派生的最大值不同的硬限制，请将 /etc/system 中的 pt_cnt 和 ptms_ptymax 设置为所需的 pty 数目。在这种情况下，ptms_pctofmem 的设置没有影响。

要将一个不同百分比的系统内存专用于 pty 支持并允许操作系统管理显式限制，请执行以下操作：

- 不要在 /etc/system 中设置 pt_cnt 或 ptms_ptymax。
- 在 /etc/system 中将 pt_pctofmem 设置为所需的百分比。例如，要设置为 10%，请设置 pt_pctofmem=10。

请注意，在内存用于支持 pty 时，才会实际分配内存。内存一旦分配，它将保持为已分配状态。

pt_cnt

说明 可用 /dev/pts 条目数是动态的，最大值是由系统中的可用物理内存量所确定的限制。pt_cnt 是用于确定系统可以容纳的最小登录数的三个变量之一。系统可以支持的缺省最大 /dev/pts 设备数是在引导时确定的，确定方法是计算在一定百分比的系统内存内可以容纳的 pty 结构数（请参见 pt_pctofmem）。如果 pt_cnt 为零，系统可以一直分配到该最大值。如果 pt_cnt 为非零值，系统可以一直分配到 pt_cnt 和缺省最大值中的较大值。

数据类型 无符号整数

缺省值 0

范围 0 - maxpid

单位 登录/窗口

动态? 否

验证 无

何时更改 当需要显式控制可以远程登录到系统中的用户数时。

交付级别 不稳定

pt_pctofmem

说明	指定数据结构可以用来支持 /dev/pts 条目的最大物理内存百分比。系统为每个 /dev/pts 条目使用 176 个字节。
数据类型	无符号整数
缺省值	5
范围	0 - 100
单位	百分比
动态?	否
验证	无
何时更改	当需要限制或增大可以登录到系统中的用户数时。值为零意味着没有远程用户可以登录到系统中。
交付级别	不稳定

pt_max_pty

说明	定义系统提供的最大 ptys 数
数据类型	无符号整数
缺省值	0 (使用系统定义的最大值)
范围	0 - MAXUINT
单位	登录/窗口
动态?	是
验证	无
隐式	应该大于或等于 pt_cnt。只有分配的 ptys 数超过 pt_cnt 的值后，才会对值进行检查。
何时更改	即使系统基于其当前配置值可以处理更多登录，但仍要对支持的登录数设置绝对上限时。
交付级别	不稳定

STREAMS 参数

nstrpush

说明	指定可以插入（推送到）STREAM 中的模块数。
数据类型	带符号整数
缺省值	9
范围	9 - 16
单位	模块
动态?	是
验证	无
何时更改	当软件供应商要求您进行修改时。当 STREAM 超出其允许的推送计数时不会显示任何消息。将会向尝试该推送的程序返回一个 <code>EINVAL</code> 值。
交付级别	不稳定

strmsgsz

说明	指定单个系统调用可以向 STREAM 传递的要放置到消息的数据部分中的最大字节数。超过此大小的 <code>write</code> 将被拆分为多条消息。有关更多信息，请参见 write(2) 。
数据类型	带符号整数
缺省值	65,536
范围	0 - 262,144
单位	字节
动态?	是
验证	无
何时更改	当 <code>putmsg</code> 调用返回 <code>ERANGE</code> 时。有关更多信息，请参见 putmsg(2) 。
交付级别	不稳定

strctlsz

说明	指定单个系统调用可以向 STREAM 传递的要放置到消息的控制部分中的最大字节数
数据类型	带符号整数
缺省值	1024
范围	0 - MAXINT
单位	字节
动态?	是
验证	无
何时更改	当软件供应商要求您进行修改时。如果 <code>putmsg(2)</code> 调用试图超出此限制，则它们会返回 <code>ERANGE</code> 。
交付级别	不稳定

System V 消息队列

System V 消息队列提供了一个消息传递接口，使用该接口，可以通过在内核中创建的队列来交换消息。Oracle Solaris 环境中提供了用来将消息入队和出队的接口。消息可以具有与之关联的类型。入队操作会将消息置于队列的末尾。出队操作将从队列中删除特定类型的第一条消息，如果未指定类型则删除第一条消息。

有关调优这些系统资源的详细信息，请参见《[Oracle Solaris 管理：Oracle Solaris Zones、Oracle Solaris 10 Zones 和资源管理](#)》中的第 6 章“资源控制（概述）”。

System V 信号

System V 信号在 Oracle Solaris OS 中提供计数信号。信号是一个计数器，用于为多个进程提供对共享数据对象的访问。除了针对信号的标准设置和释放操作，System V 信号可以具有将根据需要增大和减小的值（例如，要表示可用资源数）。System V 信号还允许同时对一组信号执行操作，以及在进程意外终止时使系统撤消该进程的最后一个操作。

System V 共享内存

System V 共享内存允许进程创建段。协作进程可以附加到内存段（根据对段的访问权限）并访问段中包含的数据。此功能是通过可装入模块实现的。`/etc/system` 文件中的条目必须包含 `shmsys:` 前缀。

DBMS 供应商使用称为**锁定共享内存** (intimate shared memory, ISM) 的一种特殊共享内存来使性能最优。当共享内存段转变为 ISM 段时，将锁定该段的内存。使用此功能，可以沿更快的 I/O 路径前进并改善内存使用情况。然后，附加到 ISM 模式下的段的所有进程将共享描述该段的大量内核资源。

segspt_minfree

说明	确定不能分配为 ISM 共享内存的系统内存页数。
数据类型	无符号长整数
缺省值	创建第一个 ISM 段时的可用系统内存的 5%
范围	0 - 50% 的物理内存
单位	页
动态?	是
验证	无。如果值太小，当内存被 ISM 段占用时，会导致系统挂起或性能严重下降。
何时更改	在具有使用 ISM 的大量物理内存的数据库服务器上，可以减小此参数的值。如果未使用 ISM 段，此参数不起作用。在具有大量内存的计算机上，最大值为 128 MB (0x4000) 就基本足够了。
交付级别	不稳定

调度

disp_rechoose_interval

说明	与先前的 <code>rechoose_interval</code> 参数类似，此参数指定在断定某个进程已与运行它的最后一个 CPU 失去所有关联之前需要经过的时间量。不同的是，能够以更细的时间增量来设置此参数。应使用此参数代替过时的 <code>rechoose_interval</code> 参数，但是如果在 <code>/etc/system</code> 文件中设置了 <code>rechoose_interval</code> 参数，仍可以接受该参数。
----	--

此时间间隔到期后，所有 CPU 都可以用来调度线程。此参数不会应用于实时类中的线程，但是将应用于所有其他调度类中的线程。

如果要更改此参数的值，请使用 `mdb` 通过下列步骤进行：

1. 将纳秒转换为无刻度时间。例如，要将一个基于 5000000 纳秒的值转换为无刻度时间，请使用以下语法：

```
# mdb -kw
.
.
.
> 0t5000000::time -u
0xb6a444
```

2. 将 `disp_rechoose_interval` 设置为无刻度时间值。例如，提供上一步中返回的值。

```
> disp_rechoose_interval /Z 0xb6a444
disp_rechoose_interval: 0x447d998 = 0xb6a444
```

3. 确认 `disp_rechoose_interval` 已设置为正确的值。例如：

```
> disp_rechoose_interval::print
0xb6a444
```

数据类型	带符号整数
缺省值	3
范围	0 - MAXINT
动态?	是
验证	无
何时更改	当高速缓存很大时，或者如果系统正在运行某个非常重要的进程或者一组进程，而这些进程看起来存在大量高速缓存未命中问题且该问题不是由数据访问模式导致的。
	在更改此参数之前，请考虑使用处理器集功能或处理器绑定。有关更多信息，请参见 psrset(1M) 或 pbind(1M) 。
交付级别	不稳定
变更历史记录	有关信息，请参见第 158 页中的“ <code>disp_rechoose_interval</code> ”。

计时器

hires_tick

说明	当设置了此参数时，将导致 Oracle Solaris OS 使用系统时钟速率 1000，而不是缺省值 100。
数据类型	带符号整数
缺省值	0
范围	0（禁用）或 1（启用）
动态?	否。导致在引导时设置新的系统计时变量。引导后不再引用。
验证	无
何时更改	当您希望超时的精度小于 10 毫秒且大于或等于 1 毫秒时。
交付级别	不稳定

timer_max

说明	指定可用的 POSIX 计时器数。
数据类型	带符号整数
缺省值	1000
范围	0 - MAXINT
动态?	否。增大该值会导致系统崩溃。
验证	无
何时更改	当系统提供的缺省计时器数不够用时。执行 <code>timer_create</code> 系统调用时，应用程序收到 <code>EAGAIN</code> 错误。
交付级别	不稳定

特定于 SPARC 系统的参数

以下参数适用于 sun4v 和 SPARC M 系列 sun4u 平台。

tsb_alloc_hiwater_factor

说明	<p>初始化 <code>tsb_alloc_hiwater</code>，对可以为转换存储缓冲区 (translation storage buffer, TSB) 分配的物理内存量设置上限，如下所述：</p> $\text{tsb_alloc_hiwater} = \text{物理内存 (字节)} / \text{tsb_alloc_hiwater_factor}$ <p>当分配给 TSB 的内存等于 <code>tsb_alloc_hiwater</code> 的值时，TSB 内存分配算法在页被取消映射时会尝试回收 TSB 内存。</p> <p>使用此因子增大 <code>tsb_alloc_hiwater</code> 的值时，请务必小心。为防止系统挂起，得出的高水位值必须大大低于 <code>swapfs_minfree</code> 和 <code>segspt_minfree</code> 的值。</p>
数据类型	整数
缺省值	32
范围	1 - MAXINIT
	<p>请注意，该因子为 1 会使所有物理内存都可以分配给 TSB，这可能会导致系统挂起。该因子太高将导致没有内存可以分配给 TSB，从而降低系统性能。</p>
动态?	是
验证	无
何时更改	如果系统的许多进程都附加到非常大的共享内存段，请更改此参数的值。大多数情况下，不需要对此变量进行调优。
交付级别	不稳定

default_tsb_size

说明	选择分配给所有进程的初始转换存储缓冲区 (translation storage buffer, TSB) 的大小。
数据类型	整数
缺省值	缺省为 0 (8 KB)，对应于 512 个条目
范围	可能的值为：

值	说明
0	8 KB

值	说明
1	16 KB
3	32 KB
4	128 KB
5	256 KB
6	512 KB
7	1 MB

动态?	是
验证	无
何时更改	一般情况下，不需要更改此值。但是，如果系统上的大部分进程具有的工作集都大于平均值，或者如果禁用了驻留集大小 (resident set size, RSS) 调整，更改此值可能会有一定好处。
交付级别	不稳定

enable_tsb_rss_sizing

说明	启用基于驻留集大小 (resident set size, RSS) 的 TSB 大小调整试探。
数据类型	布尔值
缺省值	1 (可以调整 TSB 大小)
范围	0 (TSB 保持为 <code>tsb_default_size</code>) 或 1 (可以调整 TSB 大小) 如果设置为 0，则会忽略 <code>tsb_rss_factor</code> 。
动态?	是
验证	是
何时更改	可以设置为 0 以阻止 TSB 增长。大多数情况下，此参数应保留缺省设置。
交付级别	不稳定

tsb_rss_factor

说明	控制 RSS 大小调整试探的 RSS 与 TSB 范围的比率。此系数除以 512 可以得到必须驻留在内存中的 TSB 范围所占百分比，只有满足此百分比后，才认为 TSB 的大小可以调整。
----	---

数据类型	整数
缺省值	384，得到值 75%。因此，当 TSB 处于 3/4 充满状态时，将增大其大小。请注意，某些虚拟地址通常映射到 TSB 中的同一个槽。因此，在 TSB 处于 100% 充满状态时，会发生冲突。
范围	0 至 512
动态?	是
验证	无
何时更改	如果系统由于 TSB 未命中（例如，由于 TSB 中的虚拟地址冲突）而出现大量陷阱，您可以考虑减小此值，直至为 0。 例如，将 <code>tsb_rss_factor</code> 更改为 256（即 50%）而不是 384（即 75%），在某些情况下，可以帮助消除虚拟地址冲突，但是将会使用更多内核内存，特别是在负载非常重的系统上。 可以使用 <code>trapstat - T</code> 命令监视 TSB 活动。
交付级别	不稳定

地址组参数

本节介绍了通用内存可调参数，这些参数适用于使用非统一内存体系结构 (Non-Uniform Memory Architecture, NUMA) 的任何 SPARC 或 x86 系统。

`lpg_alloc_prefer`

说明	当所请求的页大小无法立即在本地内存组中获得，但是可以从远程内存组得到满足时，控制对大内存页的分配试探。 缺省情况下，如果本地空闲内存是碎片式的，但远程空闲内存不是，则 Oracle Solaris OS 将分配远程的大页面。将此参数设置为 1，指示应该做一些额外的工作来尝试在本地分配较大内存页，例如，也许可以将较小的页进行移动，从而在本地内存组中将空出的空间接合成较大的页面。
数据类型	布尔值
缺省值	0（如果本地空闲内存是碎片式的而远程空闲内存不是，则首选远程分配）
范围	0（如果本地空闲内存是碎片式的而远程空闲内存不是，则首选远程分配）

	1 (尽可能选用本地分配, 即使本地空闲内存是碎片式的而远程空闲内存不是)
动态?	否
验证	无
何时更改	如果系统上长时间运行的程序要分配由某个程序访问的内存, 或者已知由一组程序访问的内存将要在同一地址组 (lgroup) 中运行, 则可以将此参数设置为 1。在这些情况下, 可以随着程序的长时间运行分摊页面接合操作的额外成本。 如果多个程序要跨不同的地址组共享内存, 或者如果页面只会被使用较短时间, 则可以将此参数保留为缺省值 (0)。在这些情况下, 快速分配所请求的大小要比在特定位置分配更为重要。 可以使用 <code>trapstat - T</code> 命令观察 TLB 未命中活动。
交付级别	Uncommitted (未确定)

lgrp_mem_pset_aware

说明	如果进程在某个用户处理器集内运行, 此变量将确定为该进程 随机 放置的内存是从系统中的所有 lgroup 中选择, 还是仅从该处理器集内的处理器所包含的那些 lgroup 中选择。 有关创建处理器集的更多信息, 请参见 psrset(1M) 。
数据类型	布尔值
缺省值	0, Oracle Solaris OS 从系统中的所有 lgroup 选择内存
范围	<ul style="list-style-type: none"> ▪ 0, Oracle Solaris OS 从系统中的所有 lgroup 选择内存 (缺省) ▪ 1, 尝试仅从处理器集内的处理器所包含的那些 lgroup 选择内存。如果第一次尝试失败, 则可以在任何 lgroup 中分配内存。
动态?	否
验证	无
何时更改	当处理器集用于将应用程序相互隔离时, 将此值设置为值一 (1) 可能会带来更稳定的性能。
交付级别	Uncommitted (未确定)

Oracle Solaris ZFS 可调参数

本章介绍了可能需要考虑的 ZFS 可调参数，是否设置具体取决于您的系统和应用程序要求。此外，还提供了将 ZFS 与数据库产品结合使用时的可调参数建议。

- 第 73 页中的“ZFS 调优的注意事项”
- 第 74 页中的“ZFS ARC 参数”
- 第 75 页中的“ZFS 文件级预取”
- 第 76 页中的“ZFS 设备 I/O 队列深度”
- 第 77 页中的“使用闪存存储时调优 ZFS”
- 第 80 页中的“为数据库产品进行 ZFS 调优”

何处查找可调参数信息

可调参数	参考
Oracle Solaris 内核可调参数	第 2 章, Oracle Solaris 内核可调参数
NFS 可调参数	第 4 章, NFS 可调参数
Internet 协议套件可调参数	第 5 章, Internet 协议套件可调参数

ZFS 调优的注意事项

进行 ZFS 调优之前，请注意以下事项：

- 缺省值通常是最佳值。如果存在更好的值，则应将其设置为缺省值。其他值可能有助于完成某一给定的工作负荷，但很可能会导致其他方面的性能下降。有时会造成严重损坏。

- 应用 ZFS 调优之前，应遵循 ZFS 最佳做法。这些做法是一组已证明可在各种环境下使用并预计在可预见的将来能继续使用的建议。因此，在调优之前，请确保您已经阅读并了解最佳做法。有关更多信息，请参见《Oracle Solaris 11.1 管理：ZFS 文件系统》中的第 12 章“建议的 Oracle Solaris ZFS 做法”。
- 除非另行说明，否则可调参数为全局参数，将在整个系统范围内影响 ZFS 行为。

ZFS ARC 参数

本节介绍了与 ZFS ARC 行为相关的参数。

zfs_arc_min

说明	确定 ZFS 自适应替换高速缓存 (Adaptive Replacement Cache, ARC) 的最小大小。另请参见第 74 页中的“zfs_arc_max”。
数据类型	无符号整数 (64 位)
缺省值	64 MB
范围	64 MB - zfs_arc_max
单位	字节
动态?	否
验证	是，将对范围进行验证。
何时更改	如果系统的工作负荷对内存的需求起伏波动，ZFS ARC 会在弱需求期间缓存数据，然后在强需求期间收缩。不过，ZFS 在低于 zfs_arc_min 的值时不会收缩。一般情况下，不需要更改缺省值。
交付级别	不稳定
变更历史记录	有关信息，请参见第 157 页中的“zfs_arc_min”。

zfs_arc_max

说明	确定 ZFS 自适应替换高速缓存 (Adaptive Replacement Cache, ARC) 的最大大小。另请参见第 74 页中的“zfs_arc_min”。
数据类型	无符号整数 (64 位)
缺省值	在内存少于 4 GB 的系统上为内存的 75% 在内存大于 4 GB 的系统上为 physmem 减去 1 GB
范围	64 MB - physmem

单位	字节
动态?	否
验证	是，将对范围进行验证。
何时更改	如果将来的内存需求非常大并且有明确范围，可以考虑减小此参数的值来限制 ARC 的上限，从而使它不会因内存需求而发生争用。例如，如果知道将来的工作负荷需要 20% 的内存，则可以相应设置 ARC 上限，使它消耗的内存不会超过剩余的 80% 内存。
交付级别	不稳定
变更历史记录	有关信息，请参见第 157 页中的“zfs_arc_max”。

ZFS 文件级预取

zfs_prefetch_disable

说明 此参数确定称为 `zfetch` 的文件级预取机制。该机制分析文件读取模式和预测某些读取操作，从而减少应用程序等待时间。当前行为具有两大缺点：

- 由小型读取操作构成的顺序读取模式经常命中高速缓存。在这种情况下，当前行为会占用大量 CPU 时间来尝试查找下一个要发出的 I/O，而性能更多地是由 CPU 可用性控制。
- 已经发现在某些负载下，`zfetch` 代码会限制可伸缩性。可以使用 `lockstat -I` 命令或 `er_kernel` 完成 CPU 分析，如下所述：

<http://developers.sun.com/prodtech/cc/articles/perftools.html>

可以通过设置 `/etc/system` 文件中的 `zfs_prefetch_disable` 来禁用预取。

当 `zfs_vdev_cache_size` 处于禁用状态时，设备级预取被禁用。这表示如果 `zfs_vdev_cache_size` 处于禁用状态，将不再需要调优 `vdev cache shift`。

数据类型	布尔值
缺省值	0（启用）
范围	0（启用）或 1（禁用）
动态?	是
验证	否

何时更改	如果 <code>er_kernel</code> 的结果显示 <code>zfetch_*</code> 函数占用大量时间，或者如果使用 <code>lockstat</code> 执行的锁定分析显示 <code>zfetch</code> 锁定存在争用，则应考虑禁用文件级预取。
交付级别	不稳定

ZFS 设备 I/O 队列深度

`zfs_vdev_max_pending`

说明 此参数控制每个设备待处理的最大并发 I/O 数。

数据类型 整数

缺省值 10

范围 0 - MAXINT

动态? 是

验证 否

何时更改 在存储阵列（LUN 由大量磁盘驱动器组成）中，ZFS 队列可能成为读取 IOPS 的限制因素。该行为是采取向 ZFS 存储池提供与磁盘轴数量相同的 LUN 这一最佳做法的根本原因之一。也就是说，如果以宽度为 10 个磁盘的阵列级别 RAID 组创建 LUN，则使用 5 到 10 个 LUN 构建存储池可以使 ZFS 充分管理 I/O 队列而无需设置此特定可调参数。

但是，如果未使用任何单独的意图日志，并且存储池由 JBOD 磁盘组成，则使用较小的 `zfs_vdev_max_pending` 值（如 10）可在同步写入争用磁盘资源时改善同步写入延迟。使用单独的意图日志设备可以缓解为同步密集写入型负载调优此参数的需求，因为这些同步写入操作不会与深度较大的非同步写入操作队列争用磁盘资源。

在卷仅由少数几个轴组成的情况下，调优此参数对基于 NVRAM 的存储阵列没有效果。但是，当 ZFS 中存在由大量（超过 10 个）轴组成的卷时，此参数会限制该卷所获取的读取吞吐量。原因是当每个 LUN 中的排队 I/O 数最大达到 10 或 35 时，会导致每个存储轴上的 I/O 数小于 1，这不足以让单个磁盘传送其 IOPS。此问题可能在 `iostat actv` 队列输出接近 `zfs_vdev_max_pending` 值时出现。

设备驱动程序也可能会限制每个 LUN 中未处理的 I/O 数。如果在可以处理大量并发 IOPS 的存储阵列上使用 LUN，则设备驱动程序约束可能会限制并发性。请参考您系统中使用的驱动程序的配置。例如，QLogic

ISP2200、ISP2300 和 SP212 系列 FCL HBA (qlc) 驱动程序的限制在 `/kernel/drv/qlc.conf` 中描述为执行限制参数。

交付级别 不稳定

使用闪存存储时调优 ZFS

以下信息适用于闪存 SSD、F20 PCIe 加速卡、F40 PCIe 加速卡和 F5100 闪存存储阵列。

将 ZFS 用于闪存存储时请查看以下一般注释：

- 针对 ZIL (ZFS intent log, ZFS 意图日志)，请考虑使用由具有持久性内存的控制器管理的 LUN 或低延迟磁盘（如果可用）。相比于使用闪存进行低延迟提交，此选项可能具有更显著的成本效益。日志设备的大小必须仅够保留 10 秒的最大写入吞吐量。示例包括基于存储阵列的 LUN，或连接到 HBA（具有电池保护的写入高速缓存）的磁盘。

如果没有这样的设备可供使用，请分割单独的闪存设备池以用作 ZFS 存储池中的日志设备。

- F40 和 F20 闪存加速卡包含 4 个独立的闪存模块，并将其导出至 OS。F5100 最多包含 80 个独立的闪存模块。每个闪存模块在操作系统中均显示为单个设备。OS 将 SSD 视为单个设备。可将闪存设备用作 ZFS 日志设备以降低提交延迟，尤其是用于 NFS 服务器时。例如，用作 ZFS 日志设备的闪存设备的单个闪存模块可将单个轻量线程操作的延迟降低 10 倍。可将更多的闪存设备条带化到一起，以针对大量同步操作实现更高的吞吐量。
- 应对日志设备进行镜像以实现可靠性。为实现最大程度的保护，应在单独的闪存设备上创建镜像。对于 F20 和 F40 PCIe 加速卡，通过确保镜像位于不同的物理 PCIe 卡上来实现最大程度的保护。对于 F5100 存储阵列，通过将镜像放置在单独的 F5100 设备上上来获得最大程度的保护。
- 未用作日志设备的闪存设备可用作二级高速缓存设备。这样既可用于从主磁盘存储对 IOPS 进行负载转移，也可以改善常用数据的读取延迟。

将闪存设备添加为 ZFS 日志设备或高速缓存设备

将闪存设备添加为 ZFS 日志设备或高速缓存设备时请查看以下建议。

- 可使用 `zpool add` 命令将 ZFS 日志设备或高速缓存设备添加到现有的 ZFS 存储池。使用 `zpool add` 命令时请务必小心谨慎。如果将日志设备错误地添加为常规池设备，将会要求您销毁池然后从头开始恢复。可能会从池中删除各个日志设备本身。

- 在活动存储上尝试执行此操作之前，请熟悉 `zpool add` 命令。可以使用 `zpool add -n` 选项预览配置，而不创建此配置。例如，以下错误的 `zpool add` 预览语法会尝试将某个设备添加为日志设备：

```
# zpool add -n tank c4t1d0
vdev verification failed: use -f to override the following errors:
mismatched replication level: pool uses mirror and new vdev is disk
Unable to build pool from specified devices: invalid vdev configuration
```

以下是用于将某个日志设备添加到现有池的正确 `zpool add` 预览语法：

```
# zpool add -n tank log c4t1d0
would update 'tank' to the following configuration:
  tank
    mirror
      c4t0d0
      c5t0d0
  logs
    c4t1d0
```

如果指定了多个设备，它们将条带化到一起。有关更多信息，请参见下面的示例或 [zpool\(1M\)](#)。

闪存设备 `c4t1d0` 可添加为 ZFS 日志设备：

```
# zpool add pool log c4t1d0
```

如果有 2 个闪存设备，您可以添加镜像的日志设备：

```
# zpool add pool log mirror c4t1d0 c4t2d0
```

可用的闪存设备可添加为用于进行读取的高速缓存设备。

```
# zpool add pool cache c4t3d0
```

您不能镜像高速缓存设备，它们将条带化到一起。

```
# zpool add pool cache c4t3d0 c4t4d0
```

确保闪存设备和 NVRAM 存储设备具有正确的高速缓存刷新行为

ZFS 设计为与管理磁盘级高速缓存的存储设备一起使用。通常，ZFS 会通过请求高速缓存刷新来要求存储设备确保已将数据安全地放置到稳定的存储器上。对于 JBOD 存储而言，这种方式符合预期且不会出现任何问题。对于许多基于 NVRAM 的存储阵列而言，如果此类阵列接受高速缓存刷新请求并实际做出相应的操作而不是忽略该请求，则可能会出现性能问题。一些存储阵列会应要求刷新其大型高速缓存，而不考虑 NVRAM 保护使这些高速缓存与稳定存储器一样稳定的事实。

当 `uberblock` 更新后，ZFS 会以较低的频率发出刷新（如每 5 秒）。由于刷新频率很低，所以此处不需要考虑进行调优。ZFS 还会在每次应用程序请求同步写入（`O_DSYNC`、`fsync`、NFS 提交等）时发出刷新。应用程序将等待此类刷新完成，因而会影响性能。实际上，影响就是这么大。从性能的角度来看，这抵消了采用基于 NVRAM 的存储的优点。

最近有证据表明，当闪存设备用作日志设备时，高速缓存刷新调优有助于提高闪存设备性能。当向 ZFS 公开的所有 LUN 均来自受 NVRAM 保护的存储阵列，并且有措施确保将来不会添加任何不受保护的 LUN 时，可以通过设置 `zfs_nocacheflush` 将 ZFS 调优为不发出刷新请求。如果向 ZFS 公开的一些 LUN 不受 NVRAM 保护，则此调优操作可能会导致数据丢失、应用程序级别的损坏，甚至池损坏。在某些受 NVRAM 保护的存储阵列中，高速缓存刷新命令是一个空操作，因此这种情况下的调优操作不会对性能产生任何影响。

最近的 OS 更改是：刷新请求语义能够指示存储设备忽略请求（如果它们具有适当的保护）。此更改要求修复磁盘驱动程序，以使 NVRAM 设备支持更新的语义。如果 NVRAM 设备不识别此改进，请使用以下说明来告知 Solaris OS 不要向阵列发送任何同步高速缓存命令。如果要使用这些说明，请确保所有目标 LUN 确实受 NVRAM 保护。

有时，闪存设备和 NVRAM 设备未正确地向 OS 通告它们为非设备，且无需刷新高速缓存。高速缓存刷新操作会消耗很多资源。在某些情况下，不必要的刷新会极大地影响性能。

在应用下面的调优项之前，请查看以下 `zfs_nocacheflush` 语法限制：

- 下面的调优语法可以包含在 `sd.conf` 中，但每个供应商/产品只能有一个 `sd-config-list` 条目。
- 如果需要多个设备条目，可以使用以下语法在同一行上指定多对供应商 ID 和 `sd` 调优字符串：

```
#           "012345670123456789012345","tuning    ",
sd-config-list="|-VID1-||-----PID1-----|","param1:val1, param2:val2",
               "|-VIDN-||-----PIDN-----|","param1:val1, param3:val3";
```

确保供应商 ID (VID) 字符串填充为 8 个字符，产品 ID (PID) 字符串填充为 16 个字符，如上例所示。



注意 - 设备将忽略所有高速缓存同步命令。使用时需自担风险。

1. 使用 `format` 实用程序对存储阵列中的 LUN 运行 `inquiry` 子命令。例如：

```
# format
.
.
.
Specify disk (enter its number): x
format> inquiry
```

```
Vendor: ATA
Product: Marvell
Revision: XXXX
format>
```

2. 基于您的体系结构选择以下操作之一：

- 对于 F40 闪存设备，将以下条目添加到 `/kernel/drv/sd.conf` 中。在下面的条目中，确保 "ATA" 填充为 8 个字符，"3E128-TS2-550B01" 包含 16 个字符。字符串总长度为 24 个字符。

```
sd-config-list="ATA      3E128-TS2-550B01","disksort:false, cache-non:true";
```

- 对于 F20 和 F5100 闪存设备，基于您的体系结构选择以下操作之一。在下面的条目中，"ATA" 填充为 8 个字符，"MARVELL SD88SA02" 包含 16 个字符。字符串总长度为 24 个字符。
- 许多 SPARC 体系结构—将以下条目添加到 `/kernel/drv/ssd.conf` 中：

```
ssd-config-list = "ATA      MARVELL SD88SA02","throttle-max:32, disksort:false, cache-non:true";
```

- x64 和一些 SPARC 驱动程序—将以下条目添加到 `/kernel/drv/sd.conf` 中：

```
ssd-config-list="ATA      MARVELL SD88SA02","throttle-max:32, disksort:false, cache-non:true";
```

3. 如上所示，在 `sd-config-list` 条目中，小心地添加空格，使供应商 ID (VID) 长度为 8 个字符（此处为 ATA），并使产品 ID (PID) 长度为 16 个字符（此处为 MARVELL）。
4. 重新引导系统。

可将 `zfs_nocacheflush` 调整回其缺省值 (0)，而不会对性能造成不利影响。

闪存设备的 SCSI 取消映射注意事项

Solaris 11.1 OS 中有一个问题，会导致过度调用 SCSI 取消映射例程。该特定问题会降低闪存性能。解决方法是按如下方式禁用取消映射功能：

- 在 `/etc/system` 文件中包含以下条目。

```
set zfs:zfs_unmap_ignore_size=0
```

- 重新引导系统。

为数据库产品进行 ZFS 调优

在将 ZFS 与数据库产品结合使用之前，请注意以下事项。

- 如果数据库针对 I/O 使用固定磁盘块或记录大小，请相应地设置 ZFS `recordsize` 属性以与该大小相匹配。您可以根据每个文件系统完成上述操作，即使多个文件系统可能共享一个池。
- 通过 ZFS 的写复制设计，下调 `recordsize` 是一种提高 OLTP 性能的方法，但要进行批处理报告查询。

- ZFS 会对存储在磁盘上的每个块计算校验和。这将减少数据库层额外计算数据校验和的需求。如果使用 ZFS 计算校验和而不是在数据库层中计算校验和，则可在数据返回到应用程序前捕获并修复所有差异。
- UFS 直接 I/O 可用于克服 UFS 的一些设计缺陷，并消除数据的双缓冲。在 ZFS 中不存在任何 UFS 设计缺陷，ZFS 使用 `primarycache` 和 `secondarycache` 属性管理 ARC 中的缓冲数据。请注意，使用 `secondarycache (L2ARC)` 属性改善随机读取时需要启用 `primarycache` 属性。
- 将池空间的使用率保持在 90% 以下，以便维护池性能。

为 Oracle 数据库进行 ZFS 调优

建议单一实例模式下的所有 Oracle 数据库版本都使用 ZFS。当 ZFS 作为 NFS 共享文件系统提供时，可以与 Oracle RAC 数据库结合使用。

要为 Oracle 数据库进行 ZFS 调优，请查看以下建议：

- **验证是否运行的是最新的 Solaris 发行版**
从最新的 Solaris 10 或 Solaris 11 发行版开始，将 Solaris 10 9/10 发行版作为最低起点。
- **为 ZFS 存储池创建 LUN（如果需要）**
使用存储阵列工具创建要向 ZFS 存储池提供的 LUN。或者，考虑对镜像 ZFS 存储池使用整个磁盘。有关更多信息，请参见《[Oracle Solaris 11.1 管理：ZFS 文件系统](#)》中的第 3 章“[管理 Oracle Solaris ZFS 存储池](#)”。
- **为表、索引、撤消和临时数据的数据文件创建存储池**
考虑创建一个镜像存储池以提供更高级别的数据冗余。例如：

```
# zpool status dbpool
  pool: dbpool
  state: ONLINE
    scan: none requested
config:

          NAME                                STATE      READ WRITE CKSUM
  dbpool
    mirror-0
      c0t5000C500335F95E3d0                 ONLINE    0   0   0
      c0t5000C500335F907Fd0                 ONLINE    0   0   0
    mirror-1
      c0t5000C500335BD117d0                 ONLINE    0   0   0
      c0t5000C500335DC60Fd0                 ONLINE    0   0   0

errors: No known data errors
```

对于恢复日志活动较多的数据库（如提交次数较多的典型的 OLTP 数据库），请对单独日志设备使用单独 LUN。

- 为归档日志创建存储池

如果可用，系统的内部磁盘即可处理此类负载。archivelog 文件系统也可以是 dbpool 中的文件系统。

```
# zpool create archivepool c0t5000C500335E106Bd0
```

- 使用以下原则创建 ZFS 文件系统和设置特定的文件系统属性。

使用缺省记录大小 (128 KB) 为恢复、归档、撤消和临时数据库组件创建单独的文件系统。一般规则是将包含 Oracle 数据文件的文件系统设置为 recordsize = db_block_size。对于表数据和索引组件，创建一个记录大小为 8 KB 的文件系统。同时，考虑使用 primarycache 属性向数据库文件系统提供元数据高速缓存提示。有关 ZFS 文件系统属性的更多信息，请参见《Oracle Solaris 11.1 管理：ZFS 文件系统》中的“ZFS 属性介绍”。

- 为表数据文件和索引数据文件创建文件系统，其中 recordsize 为 8 KB。使用 primarycache 的缺省值。

```
# zfs create -o recordsize=8k -o mountpoint=/my_db_path/index dbpool/index
# zfs set logbias=throughput dbpool/index
# zfs get primarycache,recordsize,logbias dbpool/index
NAME                PROPERTY          VALUE             SOURCE
dbpool/index        primarycache      all               default
dbpool/index        recordsize        8K               local
dbpool/index        logbias           throughput        local
```

- 使用缺省的 recordsize 和 primarycache 值，为临时和撤消表空间创建文件系统。

```
# zfs create -o mountpoint=/my_db_path/temp dbpool/temp
# zfs set logbias=throughput dbpool/temp
# zfs create -o mountpoint=/my_db_path/undo dbpool/undo
# zfs set logbias=throughput dbpool/undo
```

- 使用单独的日志设备为恢复日志创建一个存储池。对于恢复日志活动较多的数据库（如提交次数较多的典型的 OLTP 数据库），请使用单独的日志设备 LUN。

将磁盘分区为两个分片，小分片 s0，大小范围介于 64 到 150 MB 之间，用于单独的日志设备。分片 s1 包含剩余的磁盘空间，用于恢复日志。

```
# zpool create redopool c0t50015179594B6F11d0s1 log c0t50015179594B6F11d0s0
# zpool status redopool
pool: redopool
state: ONLINE
scan: none requested
config:
```

NAME	STATE	READ	WRITE	CKSUM
redopool	ONLINE	0	0	0
c0t50015179594B6F11d0s1	ONLINE	0	0	0
logs				
c0t50015179594B6F11d0s0	ONLINE	0	0	0

```
errors: No known data errors
```

- 为恢复池中的恢复日志创建文件系统。使用 `recordsize` 和 `primarycache` 的缺省文件系统值。

```
# zfs create -o mountpoint=/my_db_path/redo redopool/redo
# zfs set logbias=latency redopool/redo
```

- 在归档池中为归档日志文件创建文件系统，启用压缩，使用 `recordsize` 的缺省值，并将 `primarycache` 设置为 `metadata`。

```
# zfs create -o compression=on -o primarycache=metadata -o mountpoint=
/my_db_admin_path/archive archivepool/archive
# zfs get primarycache,recordsize,compressratio,compression,available,
used,quota archivepool/archive
```

NAME	PROPERTY	VALUE	SOURCE
archivepool/archive	primarycache	metadata	local
archivepool/archive	recordsize	128K	default
archivepool/archive	compressratio	1.32x	-
archivepool/archive	compression	on	local
archivepool/archive	available	40.0G	-
archivepool/archive	used	10.0G	-
archivepool/archive	quota	50G	local

- 考虑调优存储阵列 I/O 队列（对于采用 HDS 或 EMC 存储阵列的系统）

ZFS 将聚合读取和写入 I/O，并在将其发送到驱动程序级别（其处理设备）之前管理 I/O 的优先级。`zfs_vdev_max_pending` 参数定义 ZFS 可发送到任何存储池设备的最大 I/O 数。

在传统存储环境中，`ssd_max_throttle` 和 `sd_max_throttle` 参数定义驱动程序可发送到存储器的最大并发 I/O 数。通过将 `zfs_vdev_max_pending` 缺省值设置为等于 `[s]sd_max_throttle` 参数的值，可以防止 ZFS 将 I/O 排队到其他不必要的 SD 层。

如果现有环境中的 `/etc/system` 文件具有 `ssd:ssd_max_throttle` 或 `sd:ssd_max_throttle`，请将 `zfs:zfs_vdev_max_pending` 设置为相同的值。例如，如果存储阵列管理员要求以下设置：

```
set ssd:ssd_max_throttle=20
```

请按如下所示设置该参数：

```
set ssd:ssd_max_throttle=20
set zfs:zfs_vdev_max_pending=20
```

通过设置该参数，ZFS 可控制每个 LUN 队列。这意味着存储中的总待处理 I/O 数可按如下所示进行增长：

```
number of LUNs * ZFS_VDEV_MAX_PENDING
```

- 分配足够的内存和交换资源

通过将 `zfs_arc_max` 参数调整到一个较低的值，可降低 ZFS 内存消耗，但我们仍建议为数据库使用较活跃的部分置备足够的内存，以便高速缓存元数据，对于 8 KB ZFS 记录大小估计为 1.5%，对于较大或较小的记录按比例减小或增大。存放索引文件的文件系统是文件系统高速缓存的最大受益对象，因为在内存不足的情况下它将最后一个失效。`zfs_arc_max` 参数以字节为单位，接受十进制或十六进制值。以下示例可将该参数设置为 2 GB：

```
set zfs:zfs_arc_max=2147483648
or
set zfs:zfs_arc_max=0x80000000
```

为防止应用程序因内存不足而失败，必须配置一些交换空间量。出于上述目的，将交换量配置为等于系统内存总量可始终满足需要。预期不会使用此交换空间，但需要将其作为预留区。有关增加交换空间的信息，请参见《Oracle Solaris 11.1 管理：ZFS 文件系统》中的“管理 ZFS 交换和转储设备”。

- 其他 Oracle 数据库配置建议
 - 以下白皮书中的“Configuring Your Oracle Database on ZFS File Systems”：
 - <http://www.oracle.com/technetwork/server-storage/solaris/config-solaris-zfs-wp-167894.pdf>
 - 《Dynamic SGA Tuning of Oracle Database on Oracle Solaris with DISM》白皮书：
 - <http://www.oracle.com/technetwork/articles/systems-hardware-architecture/using-dynamic-intimate-memory-sparc-168402.pdf>
 - Oracle 11g 安装指南系列
 - 《Oracle Database Quick Installation Guide 11g Release 2 (11.2) for Oracle Solaris on SPARC (64-Bit)》
 - http://docs.oracle.com/cd/E11882_01/install.112/e24349/toc.htm
 - 《Oracle Database Quick Installation Guide 11g Release 2 (11.2) for Oracle Solaris on x86-64 (64-Bit)》
 - http://docs.oracle.com/cd/E11882_01/install.112/e24351/toc.htm

ZFS 与 MySQL 结合使用的注意事项

在将 ZFS 与 MySQL 结合使用之前，请注意以下事项。

- **ZFS recordsize**
 - 使 ZFS recordsize 属性与存储引擎块大小匹配，以获得更好的 OLTP 性能。
- **InnoDB**
 - 通过已知应用程序内存资源占用（例如，用于数据库应用程序），可以限定 ARC 的大小，这样，应用程序将不需要从 ZFS 高速缓存回收其所需的内存。
 - 为日志创建单独的池。
 - 为 my.cnf 文件中的数据和日志设置不同的路径。
 - 创建数据文件之前，将 InnoDB 数据文件的 ZFS recordsize 属性设置为 16K，并使用 InnoDB 日志的缺省 recordsize 值。

NFS 可调参数

本节介绍了 NFS 可调参数。

- 第 85 页中的“调优 NFS 环境”
- 第 86 页中的“NFS 模块参数”
- 第 108 页中的“rpcmod 模块参数”

何处查找可调参数信息

可调参数	参考
Oracle Solaris 内核可调参数	第 2 章，Oracle Solaris 内核可调参数
Oracle Solaris ZFS 可调参数	第 3 章，Oracle Solaris ZFS 可调参数
Internet 协议套件可调参数	第 5 章，Internet 协议套件可调参数

调优 NFS 环境

您可以在 `/etc/system` 文件中定义 NFS 参数，在引导过程中将读取该文件。每个参数都包括与其关联的内核模块的名称。有关更多信息，请参见第 18 页中的“调优 Oracle Solaris 系统”。



注意 - 在不同的发行版中，参数名称、参数所在的模块以及缺省值可能会有所变化。在进行更改或应用早期发行版中的值之前，请先检查适用于活动 SunOS 发行版的版本的文档。

NFS 模块参数

本节介绍了与 NFS 内核模块相关的参数。

nfs:nfs3_pathconf_disable_cache

说明	控制针对已挂载 NFS 版本 3 文件系统的 pathconf 信息的高速缓存。
数据类型	整数（32 位）
缺省值	0（启用高速缓存）
范围	0（启用高速缓存）或 1（禁用高速缓存）
单位	布尔值
动态?	是
验证	无
何时更改	pathconf 信息是基于每个文件缓存的。不过，如果服务器可以动态更改针对特定文件的信息，请使用该参数禁用高速缓存。客户机没有用于验证其高速缓存条目的机制。
交付级别	不稳定

nfs:nfs4_pathconf_disable_cache

说明	控制针对已挂载 NFS 版本 4 文件系统的 pathconf 信息的高速缓存。
数据类型	整数（32 位）
缺省值	0（启用高速缓存）
范围	0（启用高速缓存）或 1（禁用高速缓存）
单位	布尔值
动态?	是
验证	无
何时更改	pathconf 信息是基于每个文件缓存的。不过，如果服务器可以动态更改针对特定文件的信息，请使用该参数禁用高速缓存。客户机没有用于验证其高速缓存条目的机制。
交付级别	不稳定

nfs:nfs_allow_preepoch_time

说明	控制是否应在客户机上显示带有不正确的或负值时间戳的文件。 在过去，NFS 客户机和 NFS 服务器都不会对返回的文件时间执行任何范围检查。线上时间戳值是无符号的，且长度为 32 位。因此，所有值都合法。 64 位 Solaris 内核上的时间戳值是有符号的，且长度为 64 位。无法确定时间字段是表示完全 32 位时间还是负时间（即 1970 年 1 月 1 日前的时间）。 无法确定从 32 位转换至 64 位时，是否对时间值进行符号扩展。如果时间值确实为负数，则应进行符号扩展。不过，如果时间值确实表示完全 32 位时间值，则不应进行符号扩展。只需禁用完全 32 位时间值，即可解决该问题。
数据类型	整数（32 位）
缺省值	0（禁用 32 位时间戳）
范围	0（禁用 32 位时间戳）或 1（启用 32 位时间戳）
单位	布尔值
动态?	是
验证	无
何时更改	即使在常规操作期间，也可能会将某些文件的时间戳值设置为在将来或过去很远的时间。如果首选使用已挂载 NFS 文件系统访问这些文件，请将该参数设置为 1 以允许直接传递时间戳值（不检查值）。
交付级别	不稳定

nfs:nfs_cots_timeo

说明	控制使用面向连接的传输协议（如 TCP）的已挂载 NFS 版本 2 文件系统的缺省 RPC 超时。
数据类型	带符号整数（32 位）
缺省值	600（60 秒）
范围	0 至 $2^{31} - 1$
单位	十分之一秒
动态?	是，但是文件系统的 RPC 超时是在挂载文件系统时设置的。要影响某个特定的文件系统，请在更改该参数后先卸载然后再挂载该文件系统。

验证	无
何时更改	TCP 在确保适当地传送请求和响应方面做得很好。不过，如果在速度特别慢的网络中往返时间很长，NFS 版本 2 客户机可能会提前超时。 增大该参数可以防止客户机不恰当地超时。值的范围很大，因此过度增大该值可能会导致长时间检测不到重新传输的情况。
交付级别	不稳定

nfs:nfs3_cots_timeo

说明	控制使用面向连接的传输协议（如 TCP）的已挂载 NFS 版本 3 文件系统的缺省 RPC 超时。
数据类型	带符号整数（32 位）
缺省值	600（60 秒）
范围	0 至 $2^{31} - 1$
单位	十分之一秒
动态?	是，但是文件系统的 RPC 超时是在挂载文件系统时设置的。要影响某个特定的文件系统，请在更改该参数后先卸载然后再挂载该文件系统。
验证	无
何时更改	TCP 在确保适当地传送请求和响应方面做得很好。不过，如果在速度特别慢的网络中往返时间很长，NFS 版本 3 客户机可能会提前超时。 增大该参数可以防止客户机不恰当地超时。值的范围很大，因此过度增大该值可能会导致长时间检测不到重新传输的情况。
交付级别	不稳定

nfs:nfs4_cots_timeo

说明	控制使用面向连接的传输协议（如 TCP）的已挂载 NFS 版本 4 文件系统的缺省 RPC 超时。 NFS 版本 4 协议规范不允许通过同一 TCP 连接重新传输。因此，该参数主要控制客户机响应特定事件的速度，这些事件包括检测强制卸载操作或检测服务器故障转移到新服务器的速度。
数据类型	带符号整数（32 位）
缺省值	600（60 秒）

范围	0 至 $2^{31} - 1$
单位	十分之一秒
动态?	是，但该参数是在挂载文件系统时设置的。要影响某个特定的文件系统，请在更改该参数后先卸载然后再挂载该文件系统。
验证	无
何时更改	TCP 在确保适当地传送请求和响应方面做得很好。不过，如果在速度特别慢的网络中往返时间很长，NFS 版本 4 客户机可能会提前超时。 增大该参数可以防止客户机不恰当地超时。值的范围很大，因此过度增大该值可能会导致长时间检测不到重新传输的情况。
交付级别	不稳定

nfs:nfs_do_symlink_cache

说明	控制是否为已挂载 NFS 版本 2 文件系统缓存符号链接文件的内容。
数据类型	整数（32 位）
缺省值	1（启用高速缓存）
范围	0（禁用高速缓存）或 1（启用高速缓存）
单位	布尔值
动态?	是
验证	无
何时更改	如果服务器更改了符号链接文件的内容，而没有更新文件上的修改时间戳，或如果时间戳的粒度过大，那么可能在很长的期间内对符号链接文件内容的更改在客户机上不可见。在这种情况下，请使用该参数禁用对符号链接内容的高速缓存。执行此操作会使更改立即对客户机上运行的应用程序可见。
交付级别	不稳定

nfs:nfs3_do_symlink_cache

说明	控制是否为已挂载 NFS 版本 3 文件系统缓存符号链接文件的内容。
数据类型	整数（32 位）
缺省值	1（启用高速缓存）
范围	0（禁用高速缓存）或 1（启用高速缓存）

单位	布尔值
动态?	是
验证	无
何时更改	如果服务器更改了符号链接文件的内容，而没有更新文件上的修改时间戳，或如果时间戳的粒度过大，那么可能在很长的期间内对符号链接文件内容的更改在客户机上不可见。在这种情况下，请使用该参数禁用对符号链接内容的高速缓存。执行此操作会使更改立即对客户机上运行的应用程序可见。
交付级别	不稳定

nfs:nfs4_do_symlink_cache

说明	控制是否为已挂载 NFS 版本 4 文件系统缓存符号链接文件的内容。
数据类型	整数（32 位）
缺省值	1（启用高速缓存）
范围	0（禁用高速缓存）或 1（启用高速缓存）
单位	布尔值
动态?	是
验证	无
何时更改	如果服务器更改了符号链接文件的内容，而没有更新文件上的修改时间戳，或如果时间戳的粒度过大，那么可能在很长的期间内对符号链接文件内容的更改在客户机上不可见。在这种情况下，请使用该参数禁用对符号链接内容的高速缓存。执行此操作会使更改立即对客户机上运行的应用程序可见。
交付级别	不稳定

nfs:nfs_dynamic

说明	控制是否为使用无连接传输协议（如 UDP）的已挂载 NFS 版本 2 文件系统启用称作 动态重新传输 的功能。该功能通过监视服务器响应时间，然后调整 RPC 超时以及读取和写入传输大小，从而尝试减少重新传输。
数据类型	整数（32 位）
缺省值	1（启用）
范围	0（禁用）或 1（启用）

动态?	是，但该参数是在挂载时针对每个文件系统设置的。要影响某个特定的文件系统，请在更改该参数后先卸载然后再挂载该文件系统。
验证	无
何时更改	不要修改该参数。
交付级别	不稳定

nfs:nfs3_dynamic

说明	控制是否为使用无连接传输协议（如 UDP）的已挂载 NFS 版本 3 文件系统启用称作 动态重新传输 的功能。该功能通过监视服务器响应时间，然后调整 RPC 超时以及读取和写入传输大小，从而尝试减少重新传输。
数据类型	整数（32 位）
缺省值	0（禁用）
范围	0（禁用）或 1（启用）
单位	布尔值
动态?	是，但该参数是在挂载时针对每个文件系统设置的。要影响某个特定的文件系统，请在更改该参数后先卸载然后再挂载该文件系统。
验证	无
何时更改	不要修改该参数。
交付级别	不稳定

nfs:nfs_lookup_neg_cache

说明	控制是否针对已挂载的 NFS 版本 2 文件系统使用负名称高速缓存。负名称高速缓存记录以前查询过但未找到的文件名称。使用该高速缓存可避免在整个网络中查询已知不存在的文件名称。
数据类型	整数（32 位）
缺省值	1（启用）
范围	0（禁用）或 1（启用）
单位	布尔值
动态?	是
验证	无

何时更改	<p>为了正确执行高速缓存，在使用负向条目前必须对其进行严格验证。对于只读的已挂载文件系统，该一致性机制稍有放宽。假定服务器上的文件系统未更改或更改速度缓慢，并且允许此类更改缓慢地传播至客户机。在这种情况下，一致性机制转变成常规属性高速缓存机制。</p> <p>如果文件系统以只读方式挂载到客户机上，但是预期文件系统将在服务器上发生更改，并且客户机需要立即看到这些更改，请使用该参数禁用负高速缓存。</p> <p>如果禁用了 <code>nfs:nfs_disable_rddir_cache</code> 参数，则可能也应当禁用该参数。有关更多信息，请参见第 100 页中的“<code>nfs:nfs_disable_rddir_cache</code>”。</p>
交付级别	不稳定

nfs:nfs3_lookup_neg_cache

说明	控制是否针对已挂载的 NFS 版本 3 只读文件系统使用负名称高速缓存。该负名称高速缓存记录已查询过但未找到的文件名称。使用该高速缓存可避免在整个网络中查询已知不存在的文件名称。
数据类型	整数（32 位）
缺省值	1（启用）
范围	0（禁用）或 1（启用）
单位	布尔值
动态?	是
验证	无
何时更改	<p>为了正确执行高速缓存，在使用负向条目前必须对其进行严格验证。对于只读的已挂载文件系统，该一致性机制稍有放宽。假定服务器上的文件系统未更改或更改速度缓慢，并且允许此类更改缓慢地传播至客户机。在这种情况下，一致性机制转变成常规属性高速缓存机制。</p> <p>负高速缓存条目仅用于只读的已挂载文件系统。如果假定服务器上的文件系统未更改或更改速度缓慢，则可允许此类更改缓慢地传播至客户机。在这种情况下，一致性机制转变成常规属性高速缓存机制。</p> <p>如果文件系统以只读方式挂载到客户机上，但是预期文件系统将在服务器上发生更改，并且客户机需要立即看到这些更改，请使用该参数禁用负高速缓存。</p>

如果禁用了 `nfs:nfs_disable_rddir_cache` 参数，则可能也应当禁用该参数。有关更多信息，请参见第 100 页中的“`nfs:nfs_disable_rddir_cache`”。

交付级别 不稳定

`nfs:nfs4_lookup_neg_cache`

说明 控制是否针对已挂载的 NFS 版本 4 文件系统使用负名称高速缓存。该负名称高速缓存记录已查询过但未找到的文件名称。使用该高速缓存可避免在整个网络中查询已知不存在的文件名称。

数据类型 整数（32 位）

缺省值 1（启用）

范围 0（禁用）或 1（启用）

单位 布尔值

动态? 是

验证 无

何时更改 为了正确执行高速缓存，在使用负向条目前必须对其进行严格验证。对于只读的已挂载文件系统，该一致性机制稍有放宽。假定服务器上的文件系统未更改或更改速度缓慢，并且允许此类更改缓慢地传播至客户机。在这种情况下，一致性机制转变成常规属性高速缓存机制。

如果文件系统以只读方式挂载到客户机上，但是预期文件系统将在服务器上发生更改，并且客户机需要立即看到这些更改，请使用该参数禁用负高速缓存。

如果禁用了 `nfs:nfs_disable_rddir_cache` 参数，则可能也应当禁用该参数。有关更多信息，请参见第 100 页中的“`nfs:nfs_disable_rddir_cache`”。

交付级别 不稳定

`nfs:nfs_max_threads`

说明 控制为 NFS 版本 2 客户机执行异步 I/O 的内核线程的数量。由于 NFS 基于 RPC，并且 RPC 本身是同步的，因此，执行来自调用线程的异步 NFS 操作时需要单独的执行上下文。

	可以异步执行的操作包括：用于预读的读取、用于 <code>readdir</code> 预读的 <code>readdir</code> 、用于 <code>putpage</code> 和 <code>pageio</code> 操作的写入，以及客户机停止使用某个文件时执行的提交和非活动清除操作。
数据类型	无符号短整型
缺省值	8
范围	0 至 $2^{15} - 1$
单位	线程
动态?	是，但该参数是在挂载时针对每个文件系统设置的。要影响某个特定的文件系统，请在更改该参数后先卸载然后再挂载该文件系统。
验证	无
何时更改	要增大或减小任一时刻未处理的并发 I/O 操作数量时。例如，对于带宽很低的网络，可能需要减小该值，以免 NFS 客户机使网络过载。而对于带宽很高的网络，客户机和服务器具有足够的资源，可能需要增大该值。这样可以更加高效地利用可用网络带宽以及客户机和服务器资源。
交付级别	不稳定

nfs:nfs3_max_threads

说明	控制为 NFS 版本 3 客户机执行异步 I/O 的内核线程的数量。由于 NFS 基于 RPC，并且 RPC 本身是同步的，因此，执行来自调用线程的异步 NFS 操作时需要单独的执行上下文。
	可以异步执行的操作包括：用于预读的读取、用于 <code>readdir</code> 预读的 <code>readdir</code> 、用于 <code>putpage</code> 和 <code>pageio</code> 请求的写入以及提交。
数据类型	无符号短整型
缺省值	8
范围	0 至 $2^{15} - 1$
单位	线程
动态?	是，但该参数是在挂载时针对每个文件系统设置的。要影响某个特定的文件系统，请在更改该参数后先卸载然后再挂载该文件系统。
验证	无
何时更改	要增大或减小任一时刻未处理的并发 I/O 操作数量时。例如，对于带宽很低的网络，可能需要减小该值，以免 NFS 客户机使网络过载。而对于带宽很高的网络，客户机和服务器具有足够的资源，可能需要增大该值。这样可以更加高效地利用可用网络带宽以及客户机和服务器资源。

交付级别 不稳定

nfs:nfs4_max_threads

说明 控制为 NFS 版本 4 客户机执行异步 I/O 的内核线程的数量。由于 NFS 基于 RPC，并且 RPC 本身是同步的，因此，执行来自调用线程的异步 NFS 操作时需要单独的执行上下文。

可以异步执行的操作包括：用于预读的读取、延迟写、目录预读以及客户机停止使用某个文件时执行的清理操作。

数据类型 无符号短整型

缺省值 8

范围 0 至 $2^{15} - 1$

单位 线程

动态? 是，但该参数是在挂载时针对每个文件系统设置的。要影响某个特定的文件系统，请在更改该参数后先卸载然后再挂载该文件系统。

验证 无

何时更改 要增大或减小任一时刻未处理的并发 I/O 操作数量时。例如，对于带宽很低的网络，可能需要减小该值，以免 NFS 客户机使网络过载。而对于带宽很高的网络，客户机和服务器具有足够的资源，可能需要增大该值。这样可以更加高效地利用可用网络带宽以及客户机和服务器资源。

交付级别 不稳定

nfs:nfs_nra

说明 控制在发现对文件进行顺序存取时，由 NFS 版本 2 客户机排队的预读操作的数量。此类预读操作会增加并发性和读取吞吐量。每个预读请求通常针对文件数据的一个逻辑块。

数据类型 整数（32 位）

缺省值 4

范围 0 至 $2^{31} - 1$

单位 逻辑块。

动态? 是

验证 无

何时更改 要增大或减小任一时刻针对特定文件的未处理预读请求数量时。例如，对于带宽很低的网络或内存较低的客户机，可能需要减小该值，以免 NFS 客户机使网络或系统内存过载。而对于带宽很高的网络，客户机和服务器具有足够的资源，可能需要增大该值。这样可以更加高效地利用可用网络带宽以及客户机和服务器资源。

交付级别 不稳定

nfs:nfs3_nra

说明 控制在发现对文件进行顺序存取时，由 NFS 版本 3 客户机排队的预读操作的数量。此类预读操作会增加并发性和读取吞吐量。每个预读请求通常针对文件数据的一个逻辑块。

数据类型 整数（32 位）

缺省值 4

范围 0 至 $2^{31} - 1$

单位 逻辑块。（请参见第 101 页中的“nfs:nfs3_bsize”。）

动态? 是

验证 无

何时更改 要增大或减小任一时刻针对特定文件的未处理预读请求数量时。例如，对于带宽很低的网络或内存较低的客户机，可能需要减小该值，以免 NFS 客户机使网络或系统内存过载。而对于带宽很高的网络，客户机和服务器具有足够的资源，可能需要增大该值。这样可以更加高效地利用可用网络带宽以及客户机和服务器资源。

交付级别 不稳定

nfs:nfs4_nra

说明 控制在发现对文件进行顺序存取时，由 NFS 版本 4 客户机排队的预读操作的数量。此类预读操作会增加并发性和读取吞吐量。每个预读请求通常针对文件数据的一个逻辑块。

数据类型 整数（32 位）

缺省值 4

范围 0 至 $2^{31} - 1$

单位 逻辑块。（请参见第 101 页中的“nfs:nfs4_bsize”。）

动态?	是
验证	无
何时更改	要增大或减小任一时刻针对特定文件的未处理预读请求数量时。例如，对于带宽很低的网络或内存较低的客户机，可能需要减小该值，以免 NFS 客户机使网络或系统内存过载。而对于带宽很高的网络，客户机和服务器具有足够的资源，可能需要增大该值。这样可以更加高效地利用可用网络带宽以及客户机和服务器资源。
交付级别	不稳定

nfs:nrnode

说明	控制 NFS 客户机上 <code>nrnode</code> 高速缓存的大小。 <code>nrnode</code> 是描述 NFS 客户机上文件的核心数据结构，NFS 版本 2、3、4 客户机都使用该结构。 <code>nrnode</code> 包含标识服务器上的文件的文件句柄。 <code>nrnode</code> 还包含指向各个高速缓存的指针，NFS 客户机使用这些高速缓存来避免对服务器的网络调用。每个 <code>nrnode</code> 与 <code>vnnode</code> 一对一关联。 <code>vnnode</code> 用来高速缓存文件数据。 NFS 客户机尽量使维护的 <code>nrnode</code> 数量最少，以努力避免销毁高速缓存的数据和元数据。重用或释放 <code>nrnode</code> 时，必须销毁高速缓存的数据和元数据。
数据类型	整数（32 位）
缺省值	该参数的缺省设置为 0，这表示应将 <code>nrnode</code> 的值设置为 <code>ncsize</code> 参数的值。实际上，任何非正的 <code>nrnode</code> 值都会导致将 <code>nrnode</code> 的值设置为 <code>ncsize</code> 的值。
范围	1 至 $2^{31} - 1$
单位	<code>nrnode</code>
动态?	否。只能通过通过在 <code>/etc/system</code> 文件中添加或更改参数，然后重新引导系统来更改该值。
验证	系统强制设置一个最大值，使得 <code>nrnode</code> 高速缓存只能消耗 25% 的可用内存。
何时更改	因为 <code>nrnode</code> 是动态创建和销毁的，所以系统倾向于设置一个大小为 <code>nrnode</code> 的高速缓存，并在系统内存压力增大或同时访问的文件数量增加时自动调整高速缓存大小。不过，在特定情况下，如果可以提前预测要访问的文件组合，则您可以自己设置 <code>nrnode</code> 的值。例如，如果 NFS 客户机将访问少量非常大的文件，可以将 <code>nrnode</code> 的值设置为较小的数值，以便由系统内

存高速缓存文件数据，而不使用 `rnode`。如果客户机将访问大量小文件，则可以增大 `nrnode` 的值来优化对文件元数据的存储，从而减少对元数据的网络调用次数。

将 `nrnode` 的值设置为 1 可以有效地禁用 `rnode` 高速缓存，但不推荐这样做。该值指示客户机仅高速缓存 1 个 `rnode`，这表示将频繁重新使用它。

交付级别 不稳定

`nfs:nfs_shrinkreaddir`

说明 某些早期的 NFS 服务器可能无法正确处理目录信息超过 1024 字节的 NFS 版本 2 `READDIR` 请求。此问题是由服务器实现中的一个错误导致的。不过，该参数在 NFS 版本 2 客户机中包含了一个解决方法。

启用该参数后，客户机将不生成目录信息超过 1024 字节的 `READDIR` 请求。如果禁用该参数，则会将线上大小设置为使用 `getdents` 系统调用传递的大小和使用 `NFS_MAXDATA` 传递的大小（即 8192 字节）中的较小值。有关更多信息，请参见 [getdents\(2\)](#)。

数据类型 整数（32 位）

缺省值 0（禁用）

范围 0（禁用）或 1（启用）

单位 布尔值

动态? 是

验证 无

何时更改 如果使用的是早期的 NFS 版本 2 服务器，并且该服务器尝试读取目录时发生互操作性问题，请检查该参数的值。启用该参数可能导致读取目录的应用程序的性能稍微降低。

交付级别 不稳定

`nfs:nfs3_shrinkreaddir`

说明 某些早期的 NFS 服务器可能无法正确处理目录信息超过 1024 字节的 NFS 版本 3 `READDIR` 请求。此问题是由服务器实现中的一个错误导致的。不过，该参数在 NFS 版本 3 客户机中包含了一个解决方法。

启用该参数后，客户机将不生成目录信息超过 1024 字节的 REaddir 请求。如果禁用该参数，则会将线上大小设置为使用 `getdents` 系统调用传递的大小或使用 `MAXBSIZE` 传递的大小（即 8192 字节）中的较小值。有关更多信息，请参见 [getdents\(2\)](#)。

数据类型	整数（32 位）
缺省值	0（禁用）
范围	0（禁用）或 1（启用）
单位	布尔值
动态?	是
验证	无
何时更改	如果使用的是早期的 NFS 版本 3 服务器，并且该服务器尝试读取目录时发生互操作性问题，请检查该参数的值。启用该参数可能导致读取目录的应用程序的性能稍微降低。
交付级别	不稳定

nfs:nfs_write_error_interval

说明	控制记录 NFS 客户机收到的 <code>ENOSPC</code> 和 <code>EDQUOT</code> 写入错误的持续时间。该参数影响 NFS 版本 2、3、4 客户机。
数据类型	长整数（64 位）
缺省值	5 秒
范围	0 至 $2^{63} - 1$
单位	秒
动态?	是
验证	无
何时更改	可增大或减小该参数以应对由客户机记录的消息量。通常，当服务器上的整个文件系统都处于活跃状态时，您可能希望增大该参数值以减少输出的空间不足消息的数量。
交付级别	不稳定

nfs:nfs_write_error_to_cons_only

说明	控制是将 NFS 写入错误同时记录到系统控制台和 <code>syslog</code> 中，还是只记录到系统控制台中。该参数影响 NFS 版本 2、3、4 客户机的消息。
数据类型	整数（32 位）
缺省值	0（系统控制台和 <code>syslog</code> ）
范围	0（系统控制台和 <code>syslog</code> ）或 1（系统控制台）
单位	布尔值
动态?	是
验证	无
何时更改	要避免包含 <code>syslogd</code> 守护进程记录的消息的文件系统被填满，请检查该参数的值。启用该参数后，将仅在系统控制台上输出消息，而不会将消息复制到 <code>syslog</code> 消息文件中。
交付级别	不稳定

nfs:nfs_disable_rddir_cache

说明	控制是否使用高速缓存来保留对 <code>REaddir</code> 和 <code>REaddirplus</code> 请求的响应。该高速缓存可避免对服务器进行线上调用来检索目录信息。
数据类型	整数（32 位）
缺省值	0（启用高速缓存）
范围	0（启用高速缓存）或 1（禁用高速缓存）
单位	布尔值
动态?	是
验证	无
何时更改	如果在目录中创建或删除文件或目录时服务器不更新该目录的修改时间，因而导致互操作性问题，请检查该参数的值。症状表现为，新名称在添加到目录中后没有显示在目录列表中，或者旧名称在从目录中删除后仍不消失。 该参数控制针对已挂载的 NFS 版本 2、3、4 文件系统的高速缓存。该参数应用于所有已挂载 NFS 文件系统，无法逐个对文件系统禁用或启用高速缓存。

如果禁用该参数，还应禁用以下参数以防止 DNLC 负高速缓存中出现错误的条目。

- 第 91 页中的“nfs:nfs_lookup_neg_cache”
- 第 92 页中的“nfs:nfs3_lookup_neg_cache”
- 第 93 页中的“nfs:nfs4_lookup_neg_cache”

交付级别 不稳定

nfs:nfs3_bsize

说明 控制 NFS 版本 3 客户机使用的逻辑块大小。该块大小表示客户机需要执行 I/O 时尝试从服务器读取或写入到服务器的数据量。

数据类型 无符号整数（32 位）

缺省值 32,768 (32 KB)

范围 0 至 $2^{31} - 1$

单位 字节

动态? 是，但文件系统的块大小是在挂载该文件系统时设置的。要影响某个特定的文件系统，请在更改该参数后先卸载然后再挂载该文件系统。

验证 无。将该参数设置得过低或过高可能会导致系统异常。不要将该参数设置为低于特定平台的 `PAGESIZE` 的值。也不要将该参数值设置得过高，因为那样可能会导致系统在等待内存分配时挂起。

何时更改 尝试更改最大数据传输大小时，请检查该参数的值。更改该参数时请结合 `nfs:nfs3_max_transfer_size` 参数。如果希望使用更大的传输大小，请增大这两个参数。如果希望使用更小的传输大小，则只需减小该参数即可。

交付级别 不稳定

nfs:nfs4_bsize

说明 控制 NFS 版本 4 客户机使用的逻辑块大小。该块大小表示客户机需要执行 I/O 时尝试从服务器读取或写入到服务器的数据量。

数据类型 无符号整数（32 位）

缺省值 32,768 (32 KB)

范围 0 至 $2^{31} - 1$

单位 字节

动态?	是，但文件系统的块大小是在挂载该文件系统时设置的。要影响某个特定的文件系统，请在更改该参数后先卸载然后再挂载该文件系统。
验证	无。将该参数设置得过低或过高可能会导致系统异常。不要将该参数设置为低于特定平台的 <code>PAGESIZE</code> 的值。也不要将该参数值设置得过高，因为那样可能会导致系统在等待内存分配时挂起。
何时更改	尝试更改最大数据传输大小时，请检查该参数的值。在更改该参数时请结合 <code>nfs:nfs4_max_transfer_size</code> 参数。如果希望使用更大的传输大小，请增大这两个参数。如果希望使用更小的传输大小，则只需减小该参数即可。
交付级别	不稳定

nfs:nfs_async_clusters

说明	<p>控制 NFS 版本 2 客户机生成的异步请求的组合。四种异步请求类型分别为 <code>read-ahead</code>、<code>putpage</code>、<code>pageio</code> 和 <code>readdir-ahead</code>。客户机试图循环使用这四种不同的请求类型，力图做到公平，不偏向某一种请求类型而牺牲另一种请求类型。</p> <p>不过，某些 NFS 版本 2 服务器中的功能（如写入收集）依赖于现有 NFS 版本 2 客户机的特定行为。具体而言，该功能依赖于客户机能够几乎在同一时间发出多个 <code>WRITE</code> 请求。如果客户机每次从队列中取出一个请求，就会妨碍这一用于增强客户机性能的服务器功能。</p> <p>因此，请使用该参数控制在切换类型之前发出的每种请求类型的请求数量。</p>
数据类型	无符号整数（32 位）
缺省值	1
范围	0 至 $2^{31} - 1$
单位	异步请求
动态?	是，但文件系统的群集设置是在挂载该文件系统时设置的。要影响某个特定的文件系统，请在更改该参数后先卸载然后再挂载该文件系统。
验证	无。不过，将该参数的值设置为 0 将导致某个特定请求类型的所有已排队请求在切换到下一类型前才得到处理。这可以有效地禁用算法的公平部分。
何时更改	要增大在切换到下一类型前生成的每种类型的异步请求数量时。执行此操作可能对依赖于来自客户机的请求群的服务器功能有帮助。
交付级别	不稳定

nfs:nfs3_async_clusters

说明	控制 NFS 版本 3 客户机生成的异步请求的组合。五种异步请求类型分别为 read-ahead、putpage、pageio、readdir-ahead 和 commit。客户机试图循环使用这四种不同的请求类型，力图做到公平，不偏向某一种请求类型而牺牲另一种请求类型。 不过，某些 NFS 版本 3 服务器中的功能（如写入收集）依赖于现有 NFS 版本 3 客户机的特定行为。具体而言，该功能依赖于客户机能够几乎在同一时间发出多个 WRITE 请求。如果客户机每次从队列中取出一个请求，就会妨碍这一用于增强客户机性能的服务器功能。 因此，请使用该参数控制在切换类型之前发出的每种请求类型的请求数量。
数据类型	无符号整数（32 位）
缺省值	1
范围	0 至 $2^{31} - 1$
单位	异步请求
动态?	是，但文件系统的群集设置是在挂载该文件系统时设置的。要影响某个特定的文件系统，请在更改该参数后先卸载然后再挂载该文件系统。
验证	无。不过，将该参数的值设置为 0 将导致某个特定请求类型的所有已排队请求在切换到下一类型前才得到处理。该值可以有效地禁用算法的公平部分。
何时更改	要增大在切换到下一类型前生成的每种类型的异步请求数量时。执行此操作可能对依赖于来自客户机的操作群的服务器功能有帮助。
交付级别	不稳定

nfs:nfs4_async_clusters

说明	控制 NFS 版本 4 客户机生成的异步请求的组合。六种异步请求类型分别为 read-ahead、putpage、pageio、readdir-ahead、commit 和 inactive。客户机试图循环使用这四种不同的请求类型，力图做到公平，不偏向某一种请求类型而牺牲另一种请求类型。 不过，某些 NFS 版本 4 服务器中的功能（如写入收集）依赖于现有 NFS 版本 4 客户机的特定行为。具体而言，该功能依赖于客户机能够几乎在同一时间发出多个 WRITE 请求。如果客户机每次从队列中取出一个请求，就会妨碍这一用于增强客户机性能的服务器功能。
----	--

	因此，请使用该参数控制在切换类型之前发出的每种请求类型的请求数量。
数据类型	无符号整数（32 位）
缺省值	1
范围	0 至 $2^{31} - 1$
单位	异步请求
动态?	是，但文件系统的群集设置是在挂载该文件系统时设置的。要影响某个特定的文件系统，请在更改该参数后先卸载然后再挂载该文件系统。
验证	无。不过，将该参数的值设置为 0 将导致某个特定请求类型的所有已排队请求在切换到下一类型前才得到处理。这可以有效地禁用算法的公平部分。
何时更改	要增大在切换到下一类型前生成的每种类型的异步请求数量时。执行此操作可能对依赖于来自客户机的请求群的服务器功能有帮助。
交付级别	不稳定

nfs:nfs_async_timeout

说明	控制执行异步 I/O 请求的线程因无事可做而退出前的休眠持续时间。当没有其他请求需要执行时，每个线程将进入休眠状态。如果在计时器到期前没有新请求进入，线程将从休眠状态唤醒并退出。如果确实有请求到达，线程将被唤醒以执行请求，直至执行完所有请求为止。然后，线程进入休眠状态等待下个请求到达，或等待计时器到期。
数据类型	整数（32 位）
缺省值	6000（1 分钟表示成 60 秒 * 100Hz）
范围	0 至 $2^{31} - 1$
单位	Hz。（通常，时钟的运行频率为 100Hz。）
动态?	是
验证	无。不过，将该参数设置为非正值将导致这些线程在队列中没有请求需处理时立即退出。
何时更改	如果已明确了解系统中应用程序的行为，并可以预测异步 I/O 请求的速率，则有可能采用以下两种方式之一调整该参数以稍微优化性能： <ul style="list-style-type: none"> ■ 通过让线程更快到期，可以更快地释放内核资源 ■ 通过让线程更慢到期，可以避免产生线程创建和销毁开销
交付级别	不稳定

nfs:nacache

说明	调整访问 NFS 客户机上的文件访问高速缓存的散列队列数量。文件访问高速缓存存储用户对他们尝试访问的文件的文件访问权限。该高速缓存自身是动态分配的。不过，用来将高速缓存编入索引的散列队列是静态分配的。该算法假定每个活动文件有一个访问高速缓存条目，并且每个散列桶有四个这样的访问高速缓存条目。因此，缺省情况下，会将该参数的值设置为 <code>nrnode</code> 参数的值。
数据类型	整数（32 位）
缺省值	该参数的缺省设置为 0。该值表示应将 <code>nacache</code> 参数的值设置为 <code>nrnode</code> 参数的值。
范围	1 至 $2^{31} - 1$
单位	访问高速缓存条目
动态?	否。只能通过通过在 <code>/etc/system</code> 文件中添加或更改参数，然后重新引导系统来更改该值。
验证	无。不过，将该参数设置为负值将有可能导致系统尝试分配非常大的一组散列队列。尝试执行此操作时，系统可能会挂起。
何时更改	如果会违反基本假设（即每个文件有一个访问高速缓存条目），请检查该参数的值。对于处于分时共享模式的系统（其中多个用户几乎同时访问同一个文件），可能会发生此违规。在这种情况下，增大访问高速缓存的预期大小可能有所帮助，从而让对高速缓存的散列访问保持高效。
交付级别	不稳定

nfs:nfs3_jukebox_delay

说明	控制 NFS 版本 3 客户机在从以前的 <code>NFS3ERR_JUKEBOX</code> 请求收到错误后等待传送新请求的持续时间。由于某种原因导致文件暂时不可用时，服务器通常会返回 <code>NFS3ERR_JUKEBOX</code> 错误。该错误通常与分层存储和 CD 或磁带自动换带机有关。
数据类型	长整数（64 位）
缺省值	1000（10 秒表示为 $10 \text{ sec} * 100\text{Hz}$ ）
范围	$0 - 2^{63} - 1$ （在 64 位平台上）
单位	Hz。（通常，时钟的运行频率为 100Hz。）
动态?	是
验证	无

何时更改 检查该参数的值，可以调整该值使其符合服务器展现出的行为。如果在使该文件可用时延迟较长，可以增大该值，以减少重复重新传输导致的网络开销。减小该值可降低在发现文件变得重新可用时的延迟。

交付级别 不稳定

nfs:nfs3_max_transfer_size

说明 控制 NFS 版本 3 READ、WRITE、REaddir 或 REaddirPLUS 请求的数据部分的最大大小。该参数控制服务器返回的请求和客户机生成的请求的最大大小。

数据类型 整数（32 位）

缺省值 1,048,576 (1 MB)

范围 0 至 $2^{31} - 1$

单位 字节

动态? 是，但该参数是在挂载时针对每个文件系统设置的。要影响某个特定的文件系统，请在更改该参数后先卸载然后再挂载该文件系统。

验证 无。不过，将服务器上的最大传输大小设置为 0 可能导致客户机出现故障，或决定不尝试与服务器对话。

通过 UDP 传输使用 NFS 时，对最大传输大小也有限制。UDP 对每个数据报有 64 KB 的硬限制。64 KB 除了包括请求的数据部分外，还必须包括 RPC 标头和其他 NFS 信息。将限制设置得过高可能会导致 UDP 出错，还会导致在客户机和服务器之间发生通信故障。

何时更改 要调整通过网络传输的数据大小时。通常情况下，还应更新 `nfs:nfs3_bsize` 参数才能反映该参数中的更改。

例如，试图将传输大小增至 32 KB 以上时，请更新 `nfs:nfs3_bsize` 以反映增大后的值。否则，将不会观察到线上请求大小的更改。有关更多信息，请参见第 101 页中的“`nfs:nfs3_bsize`”。

如果要使用小于缺省传输大小的传输大小，请针对每个文件系统使用 `mount` 命令的 `-wsize` 或 `-rsize` 选项。

交付级别 不稳定

nfs:nfs4_max_transfer_size

说明	控制 NFS 版本 4 READ、WRITE、REaddir 或 REaddirPLUS 请求的数据部分的最大大小。该参数控制服务器返回的请求和客户机生成的请求的最大大小。
数据类型	整数（32 位）
缺省值	32,768 (32 KB)
范围	0 至 $2^{31} - 1$
单位	字节
动态?	是，但该参数是在挂载时针对每个文件系统设置的。要影响某个特定的文件系统，请在更改该参数后先卸载然后再挂载该文件系统。
验证	无。不过，将服务器上的最大传输大小设置为 0 可能导致客户机出现故障，或决定不尝试与服务器对话。
	通过 UDP 传输使用 NFS 时，对最大传输大小也有限制。有关针对 UDP 的最大值的更多信息，请参见第 106 页中的“nfs:nfs3_max_transfer_size”。
何时更改	要调整通过网络传输的数据大小时。通常情况下，还应更新 nfs:nfs4_bsize 参数才能反映该参数中的更改。
	例如，试图将传输大小增至 32 KB 以上时，请更新 nfs:nfs4_bsize 以反映增大后的值。否则，将不会观察到线上请求大小的更改。有关更多信息，请参见第 101 页中的“nfs:nfs4_bsize”。
	如果要使用小于缺省传输大小的传输大小，请针对每个文件系统使用 mount 命令的 -wsize 或 -rsize 选项。
交付级别	不稳定

nfs:nfs3_max_transfer_size_clts

说明	控制基于 UDP 的 NFS 版本 3 READ、WRITE、REaddir 或 REaddirPLUS 请求的数据部分的最大大小。该参数控制服务器返回的请求和客户机生成的请求的最大大小。
数据类型	整数（32 位）
缺省值	32,768 (32 KB)
范围	0 至 $2^{31} - 1$
单位	字节

动态?	是，但该参数是在挂载时针对每个文件系统设置的。要影响某个特定的文件系统，请在更改该参数后先卸载然后再挂载该文件系统。
验证	无。不过，将服务器上的最大传输大小设置为 0 可能导致客户机出现故障，或决定不尝试与服务器对话。
何时更改	不要修改该参数。
交付级别	不稳定

nfs:nfs3_max_transfer_size_cots

说明	控制基于 TCP 的 NFS 版本 3 READ、WRITE、REaddir 或 REaddirPLUS 请求的数据部分的最大大小。该参数控制服务器返回的请求和客户机生成的请求的最大大小。
数据类型	整数（32 位）
缺省值	1,048,576 字节
范围	0 至 $2^{31} - 1$
单位	字节
动态?	是，但该参数是在挂载时针对每个文件系统设置的。要影响某个特定的文件系统，请在更改该参数后先卸载然后再挂载该文件系统。
验证	无。不过，将服务器上的最大传输大小设置为 0 可能导致客户机出现故障，或决定不尝试与服务器对话。
何时更改	请勿更改该值，除非希望使用大于 1 MB 的传输大小。
交付级别	不稳定

rpcmod 模块参数

本节描述了 rpcmod 模块的 NFS 参数。

rpcmod:clnt_max_conns

说明	控制 NFS 客户机与每个 NFS 服务器通信时所用的 TCP 连接的数量。该内核 PRC 的构造使得它可以通过单个连接多路复用 RPC。不过，如果愿意，可以使用多个连接。
数据类型	整数（32 位）

缺省值	1
范围	1 至 $2^{31} - 1$
单位	连接
动态?	是
验证	无
何时更改	通常情况下，一个连接足以达到全部网络带宽。不过，如果 TCP 无法在一个流中使用网络提供的带宽，那么采用多个连接可能会增大客户机和服务器之间的吞吐量。 增大连接数会产生一定的副作用。增大连接数还会增加跟踪每个连接所需的内核资源使用。
交付级别	不稳定

rpcmod:clnt_idle_timeout

说明	控制在关闭客户机和服务器之间的连接前，允许该连接在客户机上处于空闲状态的持续时间。
数据类型	长整数（64 位）
缺省值	300,000 毫秒（5 分钟）
范围	0 至 $2^{63} - 1$
单位	毫秒
动态?	是
验证	无
何时更改	使用该参数可以更改在关闭空闲连接前允许其在客户机上存在的时间。您可能希望以较快的速率关闭连接以避免消耗系统资源。
交付级别	不稳定

rpcmod:svc_idle_timeout

说明	控制在关闭客户机和服务器之间的连接前，允许该连接在服务器上处于空闲状态的持续时间。
数据类型	长整数（64 位）
缺省值	360,000 毫秒（6 分钟）

范围	0 至 $2^{63} - 1$
单位	毫秒
动态?	是
验证	无
何时更改	可使用该参数更改在关闭空闲连接前允许其在服务器上存在的时间。您可能希望以较快的速率关闭连接以避免消耗系统资源。
交付级别	不稳定

rpcmod:svc_default_stksize

说明	设置内核 RPC 服务线程的内核栈大小。
数据类型	整数（32 位）
缺省值	缺省值为 0。此值表示将栈大小设置为系统缺省值。
范围	0 至 $2^{31} - 1$
单位	字节
动态?	是，针对所分配的所有新线程。栈大小是在创建线程时设置的。因此，对该参数的更改不会影响现有线程，但将应用于所分配的所有新线程。
验证	无
何时更改	调用深度过大可能会导致栈溢出和红色区域故障。对传输的调用深度较深，加上对本地文件系统的调用深度很深，可能导致 NFS 服务线程溢出其栈。 请将该参数设置为平台上的硬件 <code>pagesize</code> 的倍数。
交付级别	不稳定

rpcmod:maxdupreqs

说明	控制检测无连接传输上的 RPC 级重新传输的重复请求高速缓存的大小。该高速缓存是按客户机网络地址和 RPC 过程编号、程序编号、版本号以及事务 ID 索引的。该高速缓存可避免处理可能不为幂等的重新传输的请求。
数据类型	整数（32 位）
缺省值	8192
范围	1 至 $2^{31} - 1$

单位	请求
动态?	该高速缓存的大小是动态确定的，但是对该高速缓存提供快速访问的散列队列的大小是静态确定的。将高速缓存大小设置得过大可能会导致在其中搜索条目时所需搜索时间过长。 请勿将该参数的值设置为 0。该值可防止 NFS 服务器处理非幂等的请求。
验证	无
何时更改	如果 NFS 客户机遇到误报错误，请检查该参数的值。例如，如果尝试创建某个目录时失败，但实际上创建了该目录，这可能是由于服务器没有检测到重新传输的 MKDIR 请求。 高速缓存的大小应当与服务器上的负载匹配。高速缓存记录非幂等请求，因此只需跟踪所有请求中的一部分。高速缓存确实需要将信息保留足够长时间，才能检测到客户机执行的重新传输。通常，无连接传输的客户机超时相对较短，从 1 秒左右到 20 秒左右。
交付级别	不稳定

rpcmod:cotsmaxdupreqs

说明	控制检测面向连接的传输上 RPC 级重新传输的重复请求高速缓存的大小。该高速缓存是按客户机网络地址和 RPC 过程编号、程序编号、版本号以及事务 ID 索引的。该高速缓存可避免处理可能不为幂等的重新传输的请求。
数据类型	整数（32 位）
缺省值	8192
范围	1 至 $2^{31} - 1$
单位	请求
动态?	是
验证	该高速缓存的大小是动态确定的，但是对该高速缓存提供快速访问的散列队列的大小是静态确定的。将高速缓存大小设置得过大可能会导致在其中搜索条目时所需搜索时间过长。 请勿将该参数的值设置为 0。该参数可防止 NFS 服务器处理非幂等请求。
何时更改	如果 NFS 客户机遇到误报错误，请检查该参数的值。例如，如果尝试创建某个目录失败，但实际上创建了该目录，这可能是由于服务器没有检测到重新传输的 MKDIR 请求。

高速缓存的大小应当与服务器上的负载匹配。高速缓存记录非幂等请求，因此只需跟踪所有请求中的一部分。高速缓存确实需要将信息保留足够长时间，以便检测到客户机执行的重新传输。通常，面向连接的传输的客户机超时时间很长，约1分钟左右。因此，需要将条目在高速缓存中保留很长时间。

交付级别 不稳定

Internet 协议套件可调参数

本章介绍了 Internet 协议套件的各种属性。

- 第 114 页中的“IP 可调参数”
- 第 119 页中的“TCP 可调参数”
- 第 135 页中的“UDP 可调参数”
- 第 138 页中的“IPQoS 可调参数”
- 第 138 页中的“SCTP 可调参数”
- 第 149 页中的“每路由度量”

何处查找可调参数信息

可调参数	参考
Oracle Solaris 内核可调参数	第 2 章, Oracle Solaris 内核可调参数
Oracle Solaris ZFS 可调参数	第 3 章, Oracle Solaris ZFS 可调参数
NFS 可调参数	第 4 章, NFS 可调参数

调优 IP 套件参数的概述

可以使用 `ipadm` 命令设置本章中介绍的所有调优参数, 以下参数除外:

- 第 131 页中的“`ipcl_conn_hash_size`”
- 第 131 页中的“`ip_queue_worker_wait`”
- 第 118 页中的“`ip_queue_fanout`”

只能在 `/etc/system` 文件中设置这些参数。

按照以下语法使用 `ipadm` 命令设置 TCP/IP 参数:

```
# ipadm set-prop -p parameter ip|ipv4|ipv6|tcp|udp|sctp
```

例如：

```
# ipadm set-prop -p extra_priv_ports=1047 tcp
# ipadm show-prop -p extra_priv_ports tcp
PROTO PROPERTY          PERM CURRENT      PERSISTENT  DEFAULT      POSSIBLE
tcp   extra_priv_ports      rw   1047            1047        2049,4045    1-65535
```

有关更多信息，请参见 [ipadm\(1M\)](#)。

通过使用 `ndd` 命令使用以下语法设置 TCP/IP 参数：

```
# ndd -set driver parameter value
```

有关更多信息，请参见 [ndd\(1M\)](#)。

IP 套件参数验证

将检查本节中介绍的所有参数以验证其是否落在参数范围内。参数的范围随每个参数的说明提供。

Internet 请求注解 (RFC)

RFC 文档中介绍了 Internet 协议和标准规范。可以从以下站点查看 RFC：

<https://www.ietf.org/rfc.html>

在此站点中，可以通过在 "IETF Repository Retrieval"（IETF 系统信息库检索）搜索字段中输入 RFC 编号或 Internet 草案文件名来浏览 RFC 主题。

IP 可调参数

`_icmp_err_interval` and `_icmp_err_burst`

说明 控制生成 ICMP 错误消息的 IP 速率。在任意 `_icmp_err_interval` 内，IP 最多仅可生成 `_icmp_err_burst` 条 IP 错误消息。

`_icmp_err_interval` 参数可保护 IP 免受拒绝服务攻击。将该参数设置为 0 可禁用速率限制。该操作不会禁止生成错误消息。

缺省值 `_icmp_err_interval`，100 毫秒

`_icmp_err_burst`，10 条错误消息

范围	<code>_icmp_err_interval</code> : 0–99,999 毫秒 <code>_icmp_err_burst</code> : 1–99,999 条错误消息
动态?	是
何时更改	出于诊断目的需加快错误消息生成速度时。
交付级别	不稳定
变更历史记录	有关信息，请参见第 160 页中的“IP 参数名变更 (Oracle Solaris 11)”。

`_respond_to_echo_broadcast` 和 `_respond_to_echo_multicast` (`ipv4` 或 `ipv6`)

说明	控制 IP 是否响应广播 ICMPv4 回显请求或 IPv6 多播 ICMPv6 回显请求。
缺省值	1 (启用)
范围	0 (禁用) 或 1 (启用)
动态?	是
何时更改	由于安全原因不想执行该行为时可以禁用它。
交付级别	不稳定
变更历史记录	有关信息，请参见第 160 页中的“IP 参数名变更 (Oracle Solaris 11)”。

`_send_redirects` (`ipv4` 或 `ipv6`)

说明	控制 IPv4 或 IPv6 是否发送 ICMPv4 或 ICMPv6 重定向消息。
缺省值	1 (启用)
范围	0 (禁用) 或 1 (启用)
动态?	是
何时更改	由于安全原因不想执行该行为时可以禁用它。
交付级别	不稳定
变更历史记录	有关信息，请参见第 160 页中的“IP 参数名变更 (Oracle Solaris 11)”。

forwarding (ipv4 或 ipv6)

说明	控制 IPv4 或 IPv6 是否转发带有源 IPv4 路由选项或 IPv6 路由标头的数据包。
缺省值	0 (禁用或关闭)
范围	0 (禁用或 off) 或 1 (启用或 on)
动态?	是
何时更改	使该参数保持为禁用状态可以防止拒绝服务攻击。
交付级别	不稳定
变更历史记录	有关信息, 请参见第 160 页中的“IP 参数名变更 (Oracle Solaris 11)”。

ttl

说明	为 IP 关联上的传出 IPv4 数据包控制 IPv4 标头中的生存时间 (TTL) 值。
缺省值	255
范围	1 - 255
动态?	是
何时更改	一般情况下, 不需要更改此值。
交付级别	不稳定
变更历史记录	有关信息, 请参见第 160 页中的“IP 参数名变更 (Oracle Solaris 11)”。

hoplimit (ipv6)

说明	为 IP 关联上的传出 IPv6 数据包设置 IPv6 标头中的跃点限制值。
缺省值	255
范围	0 - 255
动态?	是
何时更改	一般情况下, 不需要更改此值。
交付级别	不稳定
变更历史记录	有关信息, 请参见第 160 页中的“IP 参数名变更 (Oracle Solaris 11)”。

`_addrs_per_if`

说明	定义与实际接口相关联的逻辑 IP 接口的最大数目。
缺省值	256
范围	1 - 8,192
动态?	是
何时更改	不要更改该值。如果需要更多逻辑接口，可以考虑增大该值。但是，请注意此更改可能会对 IP 性能产生负面影响。
交付级别	不稳定
变更历史记录	有关信息，请参见第 160 页中的“IP 参数名变更 (Oracle Solaris 11)”。

`hostmodel (ipv4 或 ipv6)`

说明	控制多宿主系统上 IPv4 或 IPv6 数据包的发送和接收行为。该属性可以具有以下值： <code>weak</code> 、 <code>strong</code> 和 <code>src-priority</code> 。缺省值为 <code>weak</code> 。
缺省值	<code>weak</code>
范围	<code>weak</code> 、 <code>strong</code> 或 <code>src-priority</code> <ul style="list-style-type: none"> ■ <code>weak</code> <ul style="list-style-type: none"> ■ 传出数据包 - 传出数据包的源地址不需要与传出接口上配置的地址匹配。 ■ 传入数据包 - 传入数据包的目标地址不需要与传入接口上配置的地址匹配。 ■ <code>strong</code> <ul style="list-style-type: none"> ■ 传出数据包 - 传出数据包的源地址必须与传出接口上配置的地址匹配。 ■ 传入数据包 - 传入数据包的目标地址必须与传入接口上配置的地址匹配。 ■ <code>src-priority</code> <ul style="list-style-type: none"> ■ 传出数据包 - 如果此包中有针对 IP 目标的多个路由，系统将优先使用包中的 IP 源地址与传出接口上配置的地址相匹配的路由。 如果没有此类路由，系统将退而选择最佳路由，就像在弱 ES 情况下一样。 ■ 传入数据包 - 必须在主机的任一接口上配置传入数据包的目标地址。

动态?	是
何时更改	如果计算机具有跨严格网络域（例如，防火墙或 VPN 节点）的接口，请将该参数设置为 strong 。
交付级别	不稳定
变更历史记录	有关信息，请参见第 160 页中的“IP 参数名变更 (Oracle Solaris 11)”。

ip_squeue_fanout

说明	<p>确定将 TCP/IP 连接与 squeue 相关联的模式。</p> <p>值为 0 时会将新的 TCP/IP 连接与创建该连接的 CPU 相关联。值为 1 时会将连接与属于不同 CPU 的多个 squeue 相关联。</p>
缺省值	1
范围	0 或 1
动态?	是
何时更改	<p>在某些情况下，请考虑将该参数设置为 1 以将负载分布在所有 CPU 上。例如，当 CPU 数超过 NIC 数并且单个 CPU 无法处理单个 NIC 的网络负载时，请将该参数更改为 1。</p> <p>只能在 <code>/etc/system</code> 文件中设置该属性。</p>
区域配置	只能在全局区域中设置该参数。
交付级别	不稳定

需要额外注意的 IP 可调参数

建议不要更改以下参数。

_pathmtu_interval

说明	<p>指定 IP 刷新路径最大传输单元 (PMTU) 发现信息和尝试重新发现 PMTU 的时间间隔（以毫秒为单位）。</p> <p>请参阅关于 PMTU 发现的 RFC 1191。</p>
缺省值	1200 毫秒（20 分钟）
范围	2-999999999
动态?	是

何时更改	不要更改该值。
交付级别	不稳定
变更历史记录	有关信息，请参见第 160 页中的“IP 参数名变更 (Oracle Solaris 11)”。

`_icmp_return_data_bytes (ipv4 或 ipv6)`

说明	当 IPv4 或 IPv6 发送 ICMPv4 或 ICMPv6 错误消息时，它包括导致该错误消息的数据包的 IP 标头。该参数用于控制在 ICMPv4 或 ICMPv6 错误消息中除 IPv4 或 IPv6 标头以外要包括的额外数据包字节数。
缺省值	对于 IPv4，为 64 对于 IPv6，为 1280
范围	对于 IPv4，为 8 - 6,636 对于 IPv6，为 8 - 1,280
动态?	是
何时更改	不要更改该值。在 ICMP 错误消息中包括更多信息可能有助于对网络问题进行诊断。如果需要此功能，请增大该值。
交付级别	不稳定
变更历史记录	有关信息，请参见第 160 页中的“IP 参数名变更 (Oracle Solaris 11)”。

TCP 可调参数

`_deferred_ack_interval`

说明	为非直接相连的主机指定 TCP 延迟确认 (ACK) 计时器的超时值。 请参阅 RFC 1122，4.2.3.2。
缺省值	100 毫秒
范围	1 毫秒 - 1 分钟
动态?	是
何时更改	请勿将该值增大到超过 500 毫秒。 在下列情况下需增大该值：

- 存在大于 512 字节最大段大小 (maximum segment size, MSS) 的慢速网络链路 (小于 57.6 Kbps)
- 接收多个 TCP 段的时间间隔很短时

交付级别 不稳定

变更历史记录 有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

_local_dack_interval

说明 为直接相连的主机指定 TCP 延迟确认 (ACK) 计时器的超时值。

请参阅 RFC 1122，4.2.3.2。

缺省值 50 毫秒

范围 10 毫秒 - 500 毫秒

动态? 是

何时更改 请勿将该值增大到超过 500 毫秒。

在下列情况下需增大该值：

- 存在大于 512 字节最大段大小 (maximum segment size, MSS) 的慢速网络链路 (小于 57.6 Kbps)
- 接收多个 TCP 段的时间间隔很短时

交付级别 不稳定

变更历史记录 有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

_deferred_acks_max

说明 指定在生成确认 (ACK) 前从远程目标 (非直接相连的) 接收的 TCP 段的数目。TCP 段数是以各个连接的最大段大小 (maximum segment size, MSS) 为单位测量的。如果设置为 0 或 1，则表示 ACK 没有延迟 (假设所有段的长度均为 1 MSS)。实际数值是针对每个连接动态计算的。该值是缺省的最大值。

缺省值 2

范围 0 - 16

动态? 是

何时更改	不要更改该值。在某些情况下，由于 ACK 延迟的影响造成网络通信流量突发时，请减小该值。请勿使该值小于 2。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

_local_dacks_max

说明	指定在生成确认 (ACK) 前从直接相连的目标接收的 TCP 段的最大数目。TCP 段数是以各个连接的最大段大小 (maximum segment size, MSS) 为单位测量的。如果设置为 0 或 1，则意味着没有 ACK 延迟（假设所有段的长度均为 1 MSS）。实际数值是针对每个连接动态计算的。该值是缺省的最大值。
缺省值	8
范围	0 - 16
动态?	是
何时更改	不要更改该值。在某些情况下，由于 ACK 延迟的影响造成网络通信流量突发时，请减小该值。请勿使该值小于 2。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

_wscale_always

说明	当启用了该参数时（缺省设置），TCP 始终发送带有窗口缩放选项的 SYN 段，即使窗口缩放选项值为 0。请注意，如果 TCP 接收到带有窗口缩放选项的 SYN 段，即使已禁用该参数，TCP 也会以带有窗口缩放选项的 SYN 段进行响应。此外，将根据接收窗口大小设置选项值。 有关窗口缩放选项，请参阅 RFC 1323。
缺省值	1（启用）
范围	0（禁用）或 1（启用）
动态?	是
何时更改	如果由于不支持窗口缩放选项的旧 TCP 栈导致互操作性问题，请禁用该参数。

交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

`_tstamp_always`

说明	如果设置为 1，则 TCP 始终发送带有时间戳选项的 SYN 段。请注意，如果 TCP 接收到带有时间戳选项的 SYN 段，则 TCP 将以带有时间戳选项的 SYN 段进行响应，即使该参数设置为 0。
缺省值	0（禁用）
范围	0（禁用）或 1（启用）
动态?	是
何时更改	如果无法准确测量往返时间 (RTT) 和 TCP 序列号回绕，请启用该参数。 有关启用该选项的更多原因，请参阅 RFC 1323。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

`send_buf`

说明	定义缺省发送窗口大小（以字节为单位）。有关基于各个路由设置其他值的描述，请参阅第 149 页中的“每路由度量”。另请参见第 123 页中的“ <code>max_buf</code> ”。
缺省值	49,152
范围	4,096 - 第 123 页中的“ <code>max_buf</code> ”的当前值
动态?	是
何时更改	应用程序可使用 <code>setsockopt(3XNET)</code> <code>SO_SNDBUF</code> 来更改各个连接的发送缓冲区。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

recv_buf

说明	定义缺省接收窗口大小（以字节为单位）。有关基于各个路由设置其他值的描述，请参阅第 149 页中的“每路由度量”。另请参见第 123 页中的“max_buf”和第 135 页中的“_recv_hiwat_minmss”。
缺省值	128,000
范围	2,048 - 第 123 页中的“max_buf”的当前值
动态?	是
何时更改	应用程序可使用 <code>setsockopt(3XNET)</code> <code>SO_RCVBUF</code> 来更改各个连接的接收缓冲区。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

max_buf

说明	定义发送和接收缓冲区的最大大小（以字节为单位）。该参数控制使用 <code>setsockopt(3XNET)</code> 的应用程序可以设置的发送缓冲区和接收缓冲区最大值。
缺省值	1,048,576
范围	128,000 - 1,073,741,824
动态?	是
何时更改	如果在高速网络环境中进行 TCP 连接，请将该值增大到与网络链路速度相匹配。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

_cwnd_max

说明	定义 TCP 拥塞窗口 (cwnd) 的最大值（以字节为单位）。 有关 TCP 拥塞窗口的更多信息，请参阅 RFC 1122 和 RFC 2581。
缺省值	1,048,576

范围	128 - 1,073,741,824
动态?	是
何时更改	即使应用程序使用 <code>setsockopt(3XNET)</code> 将窗口大小更改为大于 <code>_cwnd_max</code> 的值，实际使用的窗口也不会超过 <code>_cwnd_max</code> 。因此， <code>_max_buf</code> 应大于 <code>_cwnd_max</code> 。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

`_slow_start_initial`

说明	定义值为 TCP 连接最大段大小 (MSS) 的最大初始拥塞窗口 (cwnd) 大小。 有关计算初始拥塞窗口大小的方法，请参阅 RFC 2414。
缺省值	10
范围	1 - 10
动态?	是
何时更改	不要更改该值。 如果在特殊情况下，初始 cwnd 大小导致网络拥塞，请减小该值。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

`_local_slow_start_initial`

说明	定义直接连接的主机间 TCP 连接最大段大小 (MSS) 的初始拥塞窗口 (cwnd) 大小。
缺省值	10
范围	1 - 16,384
动态?	是
何时更改	如果更大的初始窗口会改善应用程序性能，请考虑增大该参数值。
交付级别	不稳定

变更历史记录 有关信息，请参见第 160 页中的“_local_slow_start_initial”。

_slow_start_after_idle

说明	在 TCP 连接空闲（未接收到段）超过一个重新传输超时 (RTO) 时段后，该连接的拥塞窗口大小（以最大段大小 (MSS) 为单位）。 有关计算初始拥塞窗口大小的方法，请参阅 RFC 2414。
缺省值	4
范围	1 - 16,384
动态?	是
何时更改	有关更多信息，请参见第 124 页中的“_slow_start_initial”。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

sack

说明	如果设置为 2，TCP 将始终发送带有选择性确认 (SACK) 允许选项的 SYN 段。如果 TCP 接收到带有 SACK 允许选项的 SYN，并且该参数设置为 1，则 TCP 将以 SACK 允许选项进行响应。如果该参数设置为 0，TCP 将不发送 SACK 允许选项，无论传入段是否包含 SACK 允许选项。 有关 SACK 选项的信息，请参阅 RFC 2018。
缺省值	2（主动启用）
范围	0（禁用）、1（被动启用）或 2（主动启用）
动态?	是
何时更改	SACK 处理可改进 TCP 重新传输性能，因此应主动启用。有时候，另一端可能会对主动启用的 SACK 选项有困惑。如果出现这种困惑，请将该值设置为 1，以便仅在传入连接允许 SACK 处理时才启用 SACK 处理。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

_rev_src_routes

说明	如果设置为 0，由于安全原因，TCP 不会反转传入连接的 IP 源路由选项。如果设置为 1，TCP 将执行常规反向源路由。
缺省值	0（禁用）
范围	0（禁用）或 1（启用）
动态?	是
何时更改	如果出于诊断目的需使用 IP 源路由，请启用 IP 源路由。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

_time_wait_interval

说明	指定 TCP 连接处于 TIME-WAIT 状态的时间（以毫秒为单位）。 有关更多信息，请参阅 RFC 1122，4.2.2.13 节。
缺省值	60,000（60 秒）
范围	1 秒 - 10 分钟
动态?	是
何时更改	请勿将该值设置为低于 60 秒。 有关更改该参数的信息，请参阅 RFC 1122，4.2.2.13 节。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

ecn

说明	控制显式拥塞通知 (Explicit Congestion Notification, ECN) 支持。 如果该参数设置为 0，TCP 将不会与支持 ECN 机制的对等点协商。 如果在初始化连接时该参数设置为 1，TCP 将不会告知对等点它支持 ECN 机制。
----	--

但是，如果对等点在 SYN 段中指出它支持 ECN 机制，则在接受新的传入连接请求时，TCP 将告知对等点它支持 ECN 机制。

如果该参数设置为 2，除了在接受连接时以 ECN 机制与对等点协商之外，TCP 还会在传出 SYN 段中指出它在进行主动传出连接时支持 ECN 机制。

有关 ECN 的信息，请参阅 RFC 3168。

缺省值	1（被动启用）
范围	0（禁用）、1（被动启用）或 2（主动启用）
动态?	是
何时更改	ECN 可帮助 TCP 更好地处理拥塞控制。但是，现有的 TCP 实现、防火墙、NAT 和其他网络设备可能会不理解该机制。这些设备不符合 IETF 标准。 由于这些设备的原因，该参数的缺省值设置为 1。在极少数情况下，被动启用仍然会导致出现问题。仅在绝对必要时才将该参数设置为 0。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

`_conn_req_max_q`

说明	为 TCP 侦听器指定等待 <code>accept(3SOCKET)</code> 接受的待处理 TCP 连接的最大缺省数目。另请参见第 128 页中的“ <code>_conn_req_max_q0</code> ”。
缺省值	128
范围	1 - 4,294,967,295
动态?	是
何时更改	对于可能会收到多个连接请求的应用程序（例如，Web 服务器），可以增大缺省值以便与传入速率相匹配。 不要将该参数增大为极大的值。待处理的 TCP 连接会消耗过多的内存。此外，如果由于待处理的 TCP 连接数过多导致应用程序无法足够快地处理大量连接请求，新的传入请求将会被拒绝。 请注意，增大 <code>_conn_req_max_q</code> 并不意味着应用程序可以处理大量的待处理 TCP 连接。应用程序可使用 <code>listen(3SOCKET)</code> 更改每个套接字的待处理 TCP 连接的最大数目。该参数是应用程序可以使用

`listen()` 设置的最大值。因此，即使该参数设置为一个很大的值，套接字的实际最大值可能远小于 `_conn_req_max_q`，具体取决于 `listen()` 中使用的值。

交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

`_conn_req_max_q0`

说明 为 TCP 侦听器指定未完成的（未完成三次握手）待处理 TCP 连接的最大数目。

有关 TCP 三次握手的更多信息，请参阅 RFC 793。另请参见第 127 页中的“`_conn_req_max_q`”。

缺省值	1,024
范围	0 - 4,294,967,295
动态?	是

何时更改 对于可能会收到过多连接请求的应用程序（例如，Web 服务器），可以增大缺省值以便与传入速率相匹配。

下面解释了 `_conn_req_max_q0` 和每个套接字待处理连接的最大数目之间的关系。

接收到连接请求时，TCP 会首先检查等待被接受的待处理 TCP 连接数（三次握手已完成）是否超出了侦听器可接受的最大值 (N)。如果连接过多，则会拒绝请求。如果连接数在允许范围内，TCP 将检查未完成的待处理 TCP 连接数是否超出了 N 和 `_conn_req_max_q0` 的总和。如果未超出，则会接受请求。否则，将放弃最早的未完成的待处理 TCP 请求。

交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

`_conn_req_min`

说明 为侦听器指定等待被接受的待处理 TCP 连接请求最大数的缺省最小值。它是应用程序可以使用的 `listen(3SOCKET)` 最大值下限。

缺省值	1
-----	---

范围	1 - 1,024
动态?	是
何时更改	对于使用 <code>listen(3SOCKET)</code> 将待处理 TCP 连接最大数目设置为很低值的应用程序，可以使用该参数。可以增大该值以匹配传入连接请求速率。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

`_rst_sent_rate_enabled`

说明	如果该参数设置为 1，则由 <code>ipadm</code> 参数 <code>_rst_sent_rate</code> 控制发送 RST 段的最大速率。如果该参数设置为 0，则发送 RST 段时不实施速率控制。
缺省值	1 (启用)
范围	0 (禁用) 或 1 (启用)
动态?	是
何时更改	该可调参数可通过限制发送 RST 段的速率来避免在 TCP 上发生拒绝服务攻击。请仅在需要严格遵守 RFC 793 时禁用该速率控制。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

`_rst_sent_rate`

说明	设置 TCP 每秒可发送的 RST 段的最大数目。
缺省值	40
范围	0 - 4,294,967,295
动态?	是
何时更改	在 TCP 环境中，可能有合理的原因要生成比缺省值允许的数目更多的 RST。在这种情况下，请增大该参数的缺省值。
交付级别	不稳定

变更历史记录 有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

smallest_anon_port

说明 该参数控制 TCP 可选择用作临时端口的最小端口号。当应用程序以指定的协议创建连接且未指定端口号时，应用程序可以使用临时端口。临时端口不会与特定的应用程序相关联。当连接关闭后，其他应用程序可以重新使用该端口号。

单位 端口号

缺省值 32,768

范围 1,024 - 65,535

动态? 是

何时更改 需要更大的临时端口范围时。

交付级别 不稳定

变更历史记录 有关信息，请参见第 160 页中的“[tcp,sctp,udp]_smallest_anon_port 和 [tcp,sctp,udp]_largest_anon_port”。

largest_anon_port

说明 该参数控制 TCP 可选择用作临时端口的最大端口号。当应用程序以指定的协议创建连接且未指定端口号时，应用程序可以使用临时端口。临时端口不会与特定的应用程序相关联。当连接关闭后，其他应用程序可以重新使用该端口号。

单位 端口号

缺省值 65,535

范围 32,768 - 65,535

动态? 是

何时更改 需要更大的临时端口范围时。

交付级别 不稳定

变更历史记录 有关信息，请参见第 160 页中的“[tcp,sctp,udp]_smallest_anon_port 和 [tcp,sctp,udp]_largest_anon_port”。

在 `/etc/system` 文件中设置的 TCP/IP 参数

以下参数只能在 `/etc/system` 文件中设置。修改该文件后，请重新引导系统。

例如，以下条目设置 `ipcl_conn_hash_size` 参数：

```
set ip:ipcl_conn_hash_size=value
```

ipcl_conn_hash_size

说明	控制 IP 使用的连接散列表的大小。缺省值 0 表示系统根据可用内存在引导时自动将该参数调整为合适的值。
数据类型	无符号整数
缺省值	0
范围	0 - 82,500
动态?	否。只能在引导时更改该参数。
何时更改	如果系统始终有成千上万个 TCP 连接，可以相应地增大该值。增大散列表大小意味着更多的内存将被固定占用，因此会减少可供用户应用程序使用的内存。
交付级别	不稳定

ip_squeue_worker_wait

说明	控制在唤醒工作线程来处理已入队到 <code>squeue</code> 的 TCP/IP 数据包时的最大延迟。 <code>squeue</code> 是 TCP/IP 内核代码用于处理 TCP/IP 数据包的序列化队列。
缺省值	10 毫秒
范围	0 - 50 毫秒
动态?	是
何时更改	如果对响应及时性要求较高，并且网络通信流量较小，请考虑调优此参数。例如，当计算机主要为交互式网络通信流量提供服务时。 缺省值通常在网络文件服务器、Web 服务器或具有充足网络通信流量的任何服务器上效果最佳。
区域配置	只能在全局区域中设置该参数。
交付级别	不稳定

需要额外注意的 TCP 参数

建议不要更改以下参数。

`_keepalive_interval`

说明 `ipadm` 参数用于设置 TCP 连接在整个系统范围内空闲后首次发出探测的时间间隔。

Solaris 支持 TCP keep-alive 机制，如 RFC 1122 所述。该机制是通过在 TCP 套接字上设置 `SO_KEEPALIVE` 套接字选项启用的。

如果为套接字启用了 `SO_KEEPALIVE`，则在 TCP 连接空闲两小时（`tcp_keepalive_interval` 参数的缺省值）之后发送第一个 keep-alive 探测。如果八分钟之后对等点未响应探测，则会中止 TCP 连接。有关更多信息，请参见第 133 页中的“`_rexmit_interval_initial`”。

您还可以在各个应用程序上使用 `TCP_KEEPALIVE_THRESHOLD` 套接字选项覆盖缺省时间间隔，以便每个应用程序在各个套接字上使用自己的时间间隔。该选项值是一个无符号整数（以毫秒为单位）。另请参见 `tcp(7P)`。

缺省值	2 小时
范围	10 秒 - 10 天
单位	无符号整数（毫秒）
动态?	是
何时更改	不要更改该值。减小该值可能会导致不必要的网络通信流量，并且可能会因暂时的网络问题增加连接提前终止的可能性。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

`_ip_abort_interval`

说明 为 TCP 连接指定缺省的重新传输超时总值。对于给定的 TCP 连接，如果 TCP 重新传输时间已达 `_ip_abort_interval`，并且在此期间未接收到来自其他端点的任何确认，TCP 将关闭该连接。

有关 TCP 重新传输超时 (retransmission timeout, RTO) 的计算，请参阅 RFC 1122，4.2.3 节。另请参见第 133 页中的“`_rexmit_interval_max`”。

缺省值	5 分钟
范围	500 毫秒 - 1193 小时
动态?	是
何时更改	不要更改该值。有关例外情况，请参见第 133 页中的“ _rexmit_interval_max ”。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

_rexmit_interval_initial

说明	为 TCP 连接指定缺省的初始重新传输超时 (RTO) 值。有关基于各个路由设置其他值的描述，请参阅第 149 页中的“每路由度量”。
缺省值	1000 毫秒
范围	1 毫秒 - 20000 毫秒
动态?	是
何时更改	不要更改该值。减小该值会导致不必要的重新传输。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

_rexmit_interval_max

说明	定义缺省的最大重新传输超时 (RTO) 值。为所有 TCP 连接计算的 RTO 都不能超过该值。另请参见第 132 页中的“ _ip_abort_interval ”。
缺省值	6000 毫秒
范围	1 毫秒 - 7200000 毫秒
动态?	是
何时更改	在普通网络环境中，不要更改该值。 在某些特殊环境中，如果连接的往返时间 (RTT) 大约为 10 秒，则可以增大该值。若更改该值，还应当更改 <code>_ip_abort_interval</code> 参数。更改 <code>_ip_abort_interval</code> 的值，使其至少为 <code>_rexmit_interval_max</code> 值的四倍。
交付级别	不稳定

变更历史记录 有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

`_rexmit_interval_min`

说明 指定缺省的最小重新传输超时 (RTO) 值。为所有 TCP 连接计算的
 •RTO 不能低于该值。另请参见第 133 页中的“`_rexmit_interval_max`”。

缺省值 200 毫秒

范围 1 毫秒 - 7200000 毫秒

动态? 是

何时更改 在普通网络环境中，不要更改该值。

TCP 的 RTO 计算应当与大多数 RTT 波动相符合。在某些极特殊的环境中，如果连接的往返时间 (RTT) 大约为 10 秒，则增大该值。若更改该值，还应当更改 `_rexmit_interval_max` 参数。更改 `_rexmit_interval_max` 的值，使其至少为 `_rexmit_interval_min` 值的八倍。

交付级别 不稳定

变更历史记录 有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

`_rexmit_interval_extra`

说明 指定增加到计算的重新传输超时 (RTO) 值的常数。

缺省值 0 毫秒

范围 0 - 7200000 毫秒

动态? 是

何时更改 不要更改该值。

当 RTO 计算无法为连接获取适当的值时，可更改该值以避免不必要的重新传输。

交付级别 不稳定

变更历史记录 有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

`_tstamp_if_wscales`

说明	如果该参数设置为 1，并且为连接启用了窗口缩放选项，则 TCP 还会为该连接启用 <code>timestamp</code> 选项。
缺省值	1（启用）
范围	0（禁用）或 1（启用）
动态?	是
何时更改	不要更改该值。通常，当在高速网络中使用 TCP 时，避免序列号回绕是很必要的。因此，您需要 <code>timestamp</code> 选项。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

`_recv_hiwat_minmss`

说明	控制缺省的最小接收窗口大小。最小值为 <code>_recv_hiwat_minmss</code> 乘以连接的最大段大小 (MSS)。
缺省值	8
范围	1 - 65,536
动态?	是
何时更改	不要更改该值。如果必须更改该值，请确保该值不小于 4。
交付级别	不稳定
变更历史记录	有关信息，请参见第 161 页中的“TCP 参数名变更 (Oracle Solaris 11)”。

UDP 可调参数

`send_buf`

说明	为 UDP 套接字定义缺省的发送缓冲区大小。有关更多信息，请参见第 136 页中的“ <code>max_buf</code> ”。
缺省值	57,344 字节
范围	1,024 - 第 136 页中的“ <code>max_buf</code> ”的当前值
动态?	是

何时更改	请注意，应用程序可使用 <code>setsockopt(3XNET)</code> <code>SO_SNDBUF</code> 为各个套接字更改该大小。通常情况下，不需要更改缺省值。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“UDP 参数名变更 (Oracle Solaris 11)”。

recv_buf

说明	为 UDP 套接字定义缺省的接收缓冲区大小。有关更多信息，请参见第 136 页中的“max_buf”。
缺省值	57,344 字节
范围	128 - 第 136 页中的“max_buf”的当前值
动态?	是
何时更改	请注意，应用程序可使用 <code>setsockopt(3XNET)</code> <code>SO_RCVBUF</code> 为各个套接字更改该大小。通常情况下，不需要更改缺省值。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“UDP 参数名变更 (Oracle Solaris 11)”。

max_buf

说明	为 UDP 套接字定义发送和接收缓冲区的最大大小。该参数控制使用 <code>getsockopt(3SOCKET)</code> 的应用程序可以设置的发送和接收缓冲区最大大小。
缺省值	2,097,152
范围	65,536 - 1,073,741,824
动态?	是
何时更改	如果是在高速网络环境中进行关联，请增大该参数的值以匹配网络链路速度。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“UDP 参数名变更 (Oracle Solaris 11)”。

smallest_anon_port

说明	该参数控制 UDP 可选择用作临时端口的最小端口号。当应用程序以指定的协议创建连接且未指定端口号时，应用程序可以使用临时端口。临时端口不会与特定的应用程序相关联。当连接关闭后，其他应用程序可以重新使用该端口号。
单位	端口号
缺省值	32,768
范围	1,024 - 65,535
动态?	是
何时更改	需要更大的临时端口范围时。
交付级别	不稳定
变更历史记录	有关信息，请参见第 160 页中的“[tcp,sctp,udp]_smallest_anon_port 和 [tcp,sctp,udp]_largest_anon_port”。

largest_anon_port

说明	该参数控制 UDP 可选择用作临时端口的最大端口号。当应用程序以指定的协议创建连接且未指定端口号时，应用程序可以使用临时端口。临时端口不会与特定的应用程序相关联。当连接关闭后，其他应用程序可以重新使用该端口号。
单位	端口号
缺省值	65,535
范围	32,768 - 65,535
动态?	是
何时更改	需要更大的临时端口范围时。
交付级别	不稳定
变更历史记录	有关信息，请参见第 160 页中的“[tcp,sctp,udp]_smallest_anon_port 和 [tcp,sctp,udp]_largest_anon_port”。

IPQoS 可调参数

`_policy_mask`

说明 在以下任何标注位置启用或禁用 IPQoS 处理：转发出站、转发入站、本地出站和本地入站。该参数是位掩码，如下所示：

未使用	未使用	未使用	未使用	转发出站	转发入站	本地出站	本地入站
X	X	X	X	0	0	0	0

任何位置中存在 1 将在该特定标注位置中屏蔽或禁用 IPQoS 处理。例如，值 `0x01` 将为所有的本地入站数据包禁用 IPQoS 处理。

缺省值 缺省值为 0，表示在所有标注位置均启用 IPQoS 处理。

范围 0 (0x00) - 15 (0x0F)。值 15 表示在所有标注位置均禁用 IPQoS 处理。

动态? 是

何时更改 要在任何标注位置启用或禁用 IPQoS 处理时。

交付级别 不稳定

SCTP 可调参数

`_max_init_retr`

说明 控制在重新发送 INIT 块时 SCTP 端点应尝试的最大次数。SCTP 端点可使用 SCTP 启动结构覆盖该值。

缺省值 8

范围 0 - 128

动态? 是

何时更改 INIT 重新传输数目取决于第 139 页中的“`_pa_max_retr`”。理想情况下，`_max_init_retr` 应小于或等于 `_pa_max_retr`。

交付级别 不稳定

变更历史记录 有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

`_pa_max_retr`

说明	为 SCTP 关联控制重新传输最大数目（包括所有路径）。当超出该数目时，SCTP 关联将终止。
缺省值	10
范围	1 - 128
动态?	是
何时更改	所有路径中的重新传输最大数目取决于路径数和每个路径的重新传输最大数目。理想情况下，应将 <code>sctp_pa_max_retr</code> 设置为所有可用路径中第 139 页中的“ <code>_pp_max_retr</code> ”的总数。如果有 3 条路径通向目标，每条路径的重新传输最大数目为 5，那么应将 <code>_pa_max_retr</code> 设置为小于或等于 15。（请参见 RFC 2960，8.2 节中的注释）
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

`_pp_max_retr`

说明	控制指定路径中的重新传输最大数目。如果超出了某条路径的该数目，则认为无法到达该路径（目标）。
缺省值	5
范围	1 - 128
动态?	是
何时更改	不要将该值更改为小于 5。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

`_cwnd_max`

说明	为 SCTP 关联控制拥塞窗口的最大值。
缺省值	1,048,576
范围	128 - 1,073,741,824
动态?	是

何时更改	即使应用程序使用 <code>setsockopt(3XNET)</code> 将窗口大小更改为大于 <code>_cwnd_max</code> 的值，实际使用的窗口也不会超过 <code>_cwnd_max</code> 。因此，第 144 页中的“ <code>max_buf</code> ”应大于 <code>_cwnd_max</code> 。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

`_ipv4_ttl`

说明	为 SCTP 关联上的传出 IPv4 数据包控制 IP 版本 4 标头中的生存时间 (TTL) 值。
缺省值	64
范围	1 - 255
动态?	是
何时更改	一般情况下，不需要更改此值。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

`_ipv6_hoplimit`

说明	为 SCTP 关联上的传出 IPv6 数据包设置 IPv6 标头中的跃点限制值。
缺省值	60
范围	0 - 255
动态?	是
何时更改	一般情况下，不需要更改此值。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

`_heartbeat_interval`

说明	计算在 HEARTBEAT 块与允许发送心跳信号的空闲目标之间的时间间隔。
----	---------------------------------------

SCTP 端点定期发送 HEARTBEAT 块来监视其对等点的空闲目标传输地址的可到达性。

缺省值	30 秒
范围	0 - 86,400 秒
动态?	是
何时更改	请参阅 RFC 2960, 8.3 节。
交付级别	不稳定
变更历史记录	有关信息, 请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

_new_secret_interval

说明	确定何时需要生成新密钥。生成的密钥用于计算 cookie 的 MAC。
缺省值	2 分钟
范围	0 - 1,440 分钟
动态?	是
何时更改	请参阅 RFC 2960, 5.1.3 节。
交付级别	不稳定
变更历史记录	有关信息, 请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

_initial_mtu

说明	确定 SCTP 数据包的初始最大发送大小 (包括 IP 标头长度)。
缺省值	1500 字节
范围	68 - 65,535
动态?	是
何时更改	如果底层链路支持大于 1500 字节的帧大小, 请增大该参数。
交付级别	不稳定
变更历史记录	有关信息, 请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

`_deferred_ack_interval`

说明	设置 SCTP 延迟确认 (ACK) 计时器的超时值（以毫秒为单位）。
缺省值	100 毫秒
范围	1 - 60,000 毫秒
动态?	是
何时更改	请参阅 RFC 2960，6.2 节。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

`_ignore_path_mtu`

说明	启用或禁用路径 MTU 发现。
缺省值	0（禁用）
范围	0（禁用）或 1（启用）
动态?	是
何时更改	如果要忽略路径中的 MTU 更改，请启用该参数。但是，如果路径 MTU 减小，该操作可能导致 IP 分段。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

`_initial_ssthresh`

说明	为对等点的目标地址设置初始慢速启动阈值。
缺省值	1,048,576
范围	1,024 - 4,294,967,295
动态?	是
何时更改	请参阅 RFC 2960，7.2.1 节。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

send_buf

说明	定义缺省的发送缓冲区大小（以字节为单位）。另请参见第 144 页中的“max_buf”。
缺省值	102,400
范围	8,192 - 第 144 页中的“max_buf”的当前值
动态?	是
何时更改	应用程序可使用 <code>setsockopt(3XNET)</code> <code>SO_SNDBUF</code> 来更改各个连接的发送缓冲区。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

_xmit_lowat

说明	控制发送窗口大小的下限。
缺省值	8,192
范围	8,192 - 1,073,741,824
动态?	是
何时更改	一般情况下，不需要更改此值。该参数用于设置要将套接字标记为“可写”发送缓冲区中所需的最小大小。如有必要，请根据第 143 页中的“send_buf”更改该参数。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

recv_buf

说明	定义缺省的接收缓冲区大小（以字节为单位）。另请参见第 144 页中的“max_buf”。
缺省值	102,400
范围	8,192 - 第 144 页中的“max_buf”的当前值
动态?	是

何时更改	应用程序可使用 <code>setsockopt(3XNET)</code> <code>SO_RCVBUF</code> 来更改各个连接的接收缓冲区。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

max_buf

说明	控制发送和接收缓冲区大小的最大值（以字节为单位）。该参数控制使用 <code>getsockopt(3SOCKET)</code> 的应用程序可以设置的发送和接收缓冲区最大大小。
缺省值	1,048,576
范围	102,400 - 1,073,741,824
动态?	是
何时更改	如果是在高速网络环境中进行关联，请增大该参数的值以匹配网络链路速度。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

_rto_min

说明	针对对等点的所有目标地址设置重新传输超时 (RTO) 下界（以毫秒为单位）。
缺省值	1,000
范围	500 - 60,000
动态?	是
何时更改	请参阅 RFC 2960，6.3.1 节。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

`_rto_max`

说明	针对对等点的所有目标地址控制重新传输超时 (RTO) 上界（以毫秒为单位）。
缺省值	60,000
范围	1,000 - 60,000,000
动态?	是
何时更改	请参阅 RFC 2960，6.3.1 节。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

`_rto_initial`

说明	针对对等点的所有目标地址控制初始重新传输超时 (RTO)（以毫秒为单位）。
缺省值	3,000
范围	1,000 - 60,000,000
动态?	是
何时更改	请参阅 RFC 2960，6.3.1 节。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

`_cookie_life`

说明	设置 cookie 的寿命（以毫秒为单位）。
缺省值	60,000
范围	10 - 60,000,000
动态?	是
何时更改	一般情况下，不需要更改此值。可根据第 145 页中的“ <code>_rto_max</code> ”更改该参数。
交付级别	不稳定

变更历史记录 有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

`_max_in_streams`

说明 控制 SCTP 关联允许的传入流的最大数目。

缺省值 32

范围 1 - 65,535

动态? 是

何时更改 请参阅 RFC 2960，5.1.1 节。

交付级别 不稳定

变更历史记录 有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

`_initial_out_streams`

说明 控制 SCTP 关联允许的传出流的最大数目。

缺省值 32

范围 1 - 65,535

动态? 是

何时更改 请参阅 RFC 2960，5.1.1 节。

交付级别 不稳定

变更历史记录 有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

`_shutack_wait_bound`

说明 控制发送 SHUTDOWN 块之后等待 SHUTDOWN ACK 的最长时间（以毫秒为单位）。

缺省值 60,000

范围 0 - 300,000

动态? 是

何时更改 一般情况下，不需要更改此值。可根据第 145 页中的“`_rto_max`”更改该参数。

交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

_maxburst

说明	设置可在一次突发中发送的段数目的限制。
缺省值	4
范围	2 - 8
动态?	是
何时更改	无需更改该参数。您可能会出于测试目的更改该参数。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

_addip_enabled

说明	启用或禁用 SCTP 动态地址重新配置。
缺省值	0 (禁用)
范围	0 (禁用) 或 1 (启用)
动态?	是
何时更改	如果需要动态地址重新配置，可以启用该参数。由于安全原因，只应出于测试目的启用该参数。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

_prsctp_enabled

说明	为 SCTP 启用或禁用部分可靠性扩展 (RFC 3758)。
缺省值	1 (启用)
范围	0 (禁用) 或 1 (启用)
动态?	是

何时更改	如果您的 SCTP 环境不支持部分可靠性，请禁用该参数。
交付级别	不稳定
变更历史记录	有关信息，请参见第 163 页中的“SCTP 参数名变更 (Oracle Solaris 11)”。

smallest_anon_port

说明	该参数控制 SCTP 可选择用作临时端口的最小端口号。当应用程序以指定的协议创建连接且未指定端口号时，应用程序可以使用临时端口。临时端口不会与特定的应用程序相关联。当连接关闭后，其他应用程序可以重新使用该端口号。
单位	端口号
缺省值	32,768
范围	1,024 - 65,535
动态?	是
何时更改	需要更大的临时端口范围时。
交付级别	不稳定
变更历史记录	有关信息，请参见第 160 页中的“ [tcp,sctp,udp]_smallest_anon_port 和 [tcp,sctp,udp]_largest_anon_port ”。

largest_anon_port

说明	该参数控制 SCTP 可选择用作临时端口的最大端口号。当应用程序以指定的协议创建连接且未指定端口号时，应用程序可以使用临时端口。临时端口不会与特定的应用程序相关联。当连接关闭后，其他应用程序可以重新使用该端口号。
单位	端口号
缺省值	65,535
范围	32,768 - 65,535
动态?	是
何时更改	需要更大的临时端口范围时。
交付级别	不稳定

变更历史记录 有关信息，请参见第 160 页中的
 “[tcp,sctp,udp]_smallest_anon_port”和
 “[tcp,sctp,udp]_largest_anon_port”。

每路由度量

可使用许多每路由度量将一些属性与 IPv4 和 IPv6 路由表条目关联起来。

例如，一个系统有两个不同的网络接口：一个快速以太网接口和一个千兆以太网接口。系统的缺省 `recv_maxbuf` 为 128,000 字节。该缺省值对于快速以太网接口来说已足够，但对于千兆以太网接口来说不足。

您可以将一个不同的缺省 TCP 接收窗口大小与千兆以太网接口路由条目关联起来，而无需增大系统的缺省 `recv_maxbuf`。通过执行该关联，经过该路由的所有 TCP 连接都将具有增大的接收窗口大小。

例如，路由表 (`netstat -rn`) 中有以下条目，假设为 IPv4：

```
192.123.123.0      192.123.123.4      U      1      4 hme0
192.123.124.0      192.123.124.4      U      1      4 ge0
default           192.123.123.1      UG     1      8
```

在该示例中，执行以下命令：

```
# route change -net 192.123.124.0 -recvpipe x
```

然后，通向 192.123.124.0 网络（位于 `ge0` 链路上）的所有连接都将使用接收缓冲区大小 `x`，而非缺省的接收窗口大小 128,000。

如果目标位于 `a.b.c.d` 网络中，并且不存在用于该网络的特定路由条目，可以添加该网络的前缀路由并更改度量。例如：

```
# route add -net a.b.c.d 192.123.123.1 -netmask w.x.y.z
# route change -net a.b.c.d -recvpipe y
```

请注意，前缀路由的网关是缺省的路由器。然后，通向该网络的所有连接均使用接收缓冲区大小 `y`。如果有多个接口，请使用 `-ifp` 参数指定要使用的接口。这样您就可以控制将哪些接口用于特定目标。要验证该度量，请使用 `route(1M) get` 命令。

系统工具参数

本章介绍了各种系统工具的大多数参数缺省值。

- 第 152 页中的 “autofs”
- 第 152 页中的 “cron”
- 第 152 页中的 “devfsadm”
- 第 152 页中的 “dhcpagent”
- 第 153 页中的 “fs”
- 第 153 页中的 “ftp”
- 第 153 页中的 “inetinit”
- 第 153 页中的 “init”
- 第 154 页中的 “ipsec”
- 第 154 页中的 “kbd”
- 第 154 页中的 “keyserv”
- 第 154 页中的 “login”
- 第 154 页中的 “mpathd”
- 第 155 页中的 “nfs”
- 第 155 页中的 “nfslogd”
- 第 155 页中的 “nss”
- 第 155 页中的 “passwd”
- 第 155 页中的 “su”
- 第 155 页中的 “syslog”
- 第 155 页中的 “tar”
- 第 156 页中的 “telnetd”
- 第 156 页中的 “utmpd”

系统缺省参数

各种系统工具的运行是由一组在启动时由各个工具读取的值控制的。用于每个工具的值可能存储在 `/etc/default` 目录中针对该工具的一个文件中，或者位于服务管理工具 (service management facility, SMF) 配置系统信息库中的一个服务实例的属性中。有关 SMF 服务和属性的更多信息，请参见《在 Oracle Solaris 11.1 中管理服务和故障》中的“管理 SMF 服务”。

有关设置电源管理属性的信息，请参见《在 Oracle Solaris 11.1 中管理系统信息、进程和性能》中的第 5 章“管理系统控制台、终端设备和电源服务（任务）”。

autofs

您可以使用 `sharectl` 命令显示或配置 SMF `autofs` 属性。例如：

```
# sharectl get autofs
timeout=600
automount_verbose=false
automountd_verbose=false
nobrowse=false
trace=0
environment=
# sharectl set -p timeout=200 autofs
```

有关详细信息，请参见 [sharectl\(1M\)](#)。

cron

该工具可用于禁用或启用 `cron` 日志记录。

devfsadm

该文件当前未使用。

dhcpgent

DHCP 的客户端使用是由 `dhcpgent` 守护进程提供的。当 `ipadm` 用于创建 DHCP 地址对象时，或者当 `ipadm` 标识被配置为从 DHCP 接收其网络配置的一个接口时，`dhcpgent` 将启动以管理该接口上的地址。

有关更多信息，请参见 [dhcpgent\(1M\)](#) 的“文件”部分中的 `/etc/default/dhcpgent` 信息。

fs

文件系统管理命令有通用部分和特定于文件系统的部分。如果没有显式地通过 `-F` 选项指定文件系统类型，将应用缺省值。该值是在该文件中指定的。有关更多信息，请参见 `default_fs(4)` 的“说明”部分。

ftp

该工具可用于将 `ls` 命令行为设置为 RFC 959 NLST 命令。缺省 `ls` 行为与在以前的 Solaris 发行版中相同。

有关详细信息，请参见 `ftp(4)`。

inetinit

可以使用该工具配置 TCP 序列号，以及启用或禁用对 6to4 中继路由器的支持。

init

系统初始化属性现在是以下 SMF 服务的一部分：

```
svc:/system/environment:init
```

您可以使用类似以下内容的语法显示和配置系统初始化属性，例如 `TZ` 和 `LANG`。

```
# svccfg -s svc:/system/environment:init
svc:/system/environment:init> setprop
Usage: setprop pg/name = [type:] value
       setprop pg/name = [type:] ([value...])
```

Set the pg/name property of the currently selected entity. Values may be enclosed in double-quotes. Value lists may span multiple lines.

```
svc:/system/environment:init> listprop
umask                                application
umask/value_authorization           astring                               solaris.smf.value.environment
umask/umask                          astring                               022
upgrade                              application
upgrade/skip_init_upgrade            boolean                               false
upgrade/value_authorization          astring                               solaris.smf.value.environment
environment                          application
environment/LANG                    astring                               C
.
.
.
```

有关更多信息，请参见 `init(1M)` 的“文件”部分。

ipsec

可以使用该工具配置诸如 IKE 守护进程调试信息和 `ikeadm` 特权级别之类的参数。

kbd

键盘配置属性现在是以下 SMF 服务的一部分：

```
svc:/system/keymap:default
```

您可以使用类似以下内容的语法显示和配置键盘属性。

```
# svccfg -s svc:/system/keymap:default
svc:/system/keymap:default> setprop
Usage:  setprop pg/name = [type:] value
        setprop pg/name = [type:] ([value...])
```

Set the pg/name property of the currently selected entity. Values may be enclosed in double-quotes. Value lists may span multiple lines.

```
svc:/system/keymap:default> listprop
general                               framework
general/complete                      astring
general/enabled                       boolean    false
keymap                                 system
keymap/console_beeper_freq            integer    900
keymap/kbd_beeper_freq                integer    2000
keymap/keyboard_abort                 astring    enable
keymap/keyclick                       boolean    false
.
.
.
```

有关更多信息，请参见 [kbd\(1\)](#)。

keyserv

有关详细信息，请参见 [keyserv\(1M\)](#) 的“文件”部分中的 `/etc/default/keyserv` 信息。

login

有关详细信息，请参见 [login\(1\)](#) 的“文件”部分中的 `/etc/default/login` 信息。

mpathd

可使用该工具设置 `in.mpathd` 配置参数。

有关详细信息，请参见 [in.mpathd\(1M\)](#)。

nfs

您可以使用 `sharectl` 命令显示或配置 SMF NFS 属性。例如：

```
# sharectl get nfs
servers=1024
lockd_listen_backlog=32
lockd_servers=1024
lockd_retransmit_timeout=5
grace_period=90
server_versmin=2
server_versmax=4
client_versmin=2
client_versmax=4
server_delegation=on
nfsmapid_domain=
# sharectl set -p grace_period=60 nfs
```

有关详细信息，请参见 [nfs\(4\)](#)。

nfslogd

有关详细信息，请参见 [nfslogd\(1M\)](#) 的“说明”部分。

nss

可使用该工具配置 [initgroups\(3C\)](#) 查找参数。

有关详细信息，请参见 [nss\(4\)](#)。

passwd

有关详细信息，请参见 [passwd\(1\)](#) 的“文件”部分中的 `/etc/default/passwd` 信息。

su

有关详细信息，请参见 [su\(1M\)](#) 的“文件”部分中的 `/etc/default/su` 信息。

syslog

有关详细信息，请参见 [syslogd\(1M\)](#) 的“文件”部分中的 `/etc/default/syslogd` 信息。

tar

有关 `-f` 函数修饰符的说明，请参见 [tar\(1\)](#)。

如果没有提供 TAPE 环境变量且其中一个参数值为数字，并且未指定 `-f` 选项，则会在 `/etc/default/tar` 文件中查找与 `archiveN` 字符串匹配的数字。 `archiveN` 字符串用作输出设备（使用文件中指定的块和大小设置）。

例如：

```
% tar -c 2 /tmp/*
```

该命令将输出写入到在 `/etc/default/tar` 文件中指定为 `archive2` 的设备。

telnetd

该文件标识基于 telnet 连接显示的缺省 BANNER。

utmpd

utmpd 守护进程监视 `/var/adm/utmpx`（在早期 Solaris 版本中为 `/var/adm/utmp`）以确保通过 `pututxline(3C)` 由非 root 进程插入的 utmp 条目在进程终止时会被清除。

`/etc/default/utmpd` 中的两个条目受支持：

- `SCAN_PERIOD` - 在为查看被监视进程是否仍然处于活动状态而对 `/proc` 进行的各次检查之间，utmpd 休眠的秒数。缺省值为 300。
- `MAX_FDS` - utmpd 尝试监视的最大进程数。缺省值为 4096，永不需要更改。

可调参数变更历史记录

本章介绍了特定可调参数的变更历史记录。如果本节中列出了某个参数，则它自上一发行版以来已进行了变更。此外还列出了其功能已删除的参数。

- 第 157 页中的“内核参数”
- 第 160 页中的“TCP/IP 可调参数（Oracle Solaris 11 和 Oracle Solaris 11.1）”
- 第 164 页中的“已过时或已删除的参数（Oracle Solaris 11 和 Oracle Solaris 11.1）”

内核参数

常规内核及内存参数（ Oracle Solaris 11 和 Oracle Solaris 11.1 ）

zfs_arc_min

此参数说明是 Oracle Solaris 11 发行版中新增的内容。有关更多信息，请参见第 74 页中的“zfs_arc_min”。

在 Oracle Solaris 11.1 中，zfs_arc_min 参数信息已移至第 3 章，Oracle Solaris ZFS 可调参数。

zfs_arc_max

此参数说明是 Oracle Solaris 11 发行版中新增的内容。有关更多信息，请参见第 74 页中的“zfs_arc_max”。

在 Oracle Solaris 11.1 中，zfs_arc_max 参数信息已移至第 3 章，Oracle Solaris ZFS 可调参数。

disp_rechoose_interval

此参数是 Oracle Solaris 11 发行版中的新增参数。有关更多信息，请参见第 66 页中的“[disp_rechoose_interval](#)”。

分页相关参数 (Oracle Solaris 11)

fastscan

对 fastscan 的缺省值进行了阐明。有关更多信息，请参见第 43 页中的“[fastscan](#)”。

进程大小调整可调参数 (Oracle Solaris 11.1)

ngroups_max

此参数是 Oracle Solaris 11 发行版中新记录的参数。

在 Oracle Solaris 11.1 发行版中，“何时更改”说明已进行修改，包括此参数设置为大于 16 的值时用于 AUTH_SYS 验证的交互信息。有关更多信息，请参见第 36 页中的“[ngroups_max](#)”。

max_nprocs

此参数的缺省值在 Oracle Solaris 11.1 发行版中进行了更新。有关更多信息，请参见第 35 页中的“[max_nprocs](#)”。

maxusers

此参数的缺省值在 Oracle Solaris 11.1 发行版中进行了更新。有关更多信息，请参见第 33 页中的“[maxusers](#)”。

pidmax

此参数的缺省值在 Oracle Solaris 11.1 发行版中进行了更新。有关更多信息，请参见第 34 页中的“[pidmax](#)”。

segkpsize

此参数的缺省值在 Oracle Solaris 11.1 发行版中进行了更新。有关更多信息，请参见第 28 页中的“[segkpsize](#)”。

与交换相关的参数 (Oracle Solaris 11.1)

swapfs_minfree

此参数信息中的“何时更改”部分在 Oracle Solaris 11.1 发行版中进行了更新。有关更多信息，请参见第 47 页中的“[swapfs_minfree](#)”。

常规文件系统参数 (Oracle Solaris 11.1)

dnlc_dircache_percent

此参数是 Oracle Solaris 11.1 发行版中的新增参数。有关更多信息，请参见第 59 页中的“[dnlc_dircache_percent](#)”。

常规驱动程序参数 (Oracle Solaris 11)

ddi_msix_alloc_limit

此参数是 Oracle Solaris 11 发行版中新记录的参数。有关更多信息，请参见第 51 页中的“[ddi_msix_alloc_limit](#)”。

网络驱动程序参数 (Oracle Solaris 11)

igb 参数

Oracle Solaris 11 发行版中提供了 igb 网络驱动程序参数。有关更多信息，请参见第 51 页中的“[igb 参数](#)”。

ixgbe 参数

Oracle Solaris 11 发行版中提供了 ixgbe 网络驱动程序参数。有关更多信息，请参见第 52 页中的“[ixgbe 参数](#)”。

TCP/IP 可调参数 (Oracle Solaris 11 和 Oracle Solaris 11.1)

[tcp,sctp,udp]_smallest_anon_port 和 [tcp,sctp,udp]_largest_anon_port

这些参数是 Oracle Solaris 11 发行版中新记录的参数。

- 第 148 页中的 “smallest_anon_port”
- 第 148 页中的 “largest_anon_port”
- 第 130 页中的 “smallest_anon_port”
- 第 130 页中的 “largest_anon_port”
- 第 137 页中的 “smallest_anon_port”
- 第 137 页中的 “largest_anon_port”

_local_slow_start_initial

此参数是 Oracle Solaris 11.1 发行版中新记录的参数。有关更多信息，请参见第 124 页中的 “_local_slow_start_initial”。

IP 参数名变更 (Oracle Solaris 11)

以下 IP 参数已重命名为 IP 属性。

可以使用类似于以下内容的语法设置 IP 属性：

```
# ipadm set-prop -p _icmp_err_interval=100 ip
```

可以使用类似于以下内容的语法显示 IP 属性信息：

```
# ipadm show-prop -p _icmp_err_interval ip
PROTO PROPERTY          PERM CURRENT    PERSISTENT  DEFAULT  POSSIBLE
ip   _icmp_err_interval  rw   100          100         100      0-99999
```

表 A-1 IP 参数名变更

以前的 IP 参数名	IP 属性名
ip_addrs_per_if	_addrs_per_if
ip_forwarding	forwarding (IPv4)
ip6_forwarding	forwarding (IPv6)
ip_forward_src_routed	_forward_src_routed (IPv4)

表 A-1 IP 参数名变更 (续)

以前的 IP 参数名	IP 属性名
ip6_forward_src_routed	_forward_src_routed (IPv6)
ip_icmp_err_interval	_icmp_err_interval
ip_icmp_err_burst	_icmp_err_burst
ip_icmp_return_data_bytes	_icmp_return_data_bytes (IPv4)
ip6_icmp_return_data_bytes	_icmp_return_data_bytes (IPv6)
ip_ire_pathmtu_interval	_pathmtu_interval
ip_respond_to_echo_broadcast	_respond_to_echo_broadcast (IPv4)
ip6_respond_to_echo_broadcast	_respond_to_echo_broadcast (IPv6)
ip_respond_to_echo_multicast	_respond_to_echo_multicast (IPv4)
ip6_respond_to_echo_multicast	_respond_to_echo_multicast (IPv6)
ip_send_redirects	_send_redirects (IPv4)
ip6_send_redirects	_send_redirects (IPv6)
ip_strict_dst_multihoming	hostmodel

TCP 参数名变更 (Oracle Solaris 11)

以下 TCP 参数已重命名为 TCP 属性。

可以使用类似于以下内容的语法设置 TCP 属性：

```
# ipadm set-prop -p _deferred_ack_interval=100 tcp
```

可以使用类似于以下内容的语法显示 TCP 属性信息：

```
# ipadm show-prop -p _deferred_ack_interval tcp
PROTO PROPERTY          PERM CURRENT    PERSISTENT  DEFAULT  POSSIBLE
tcp    _deferred_ack_interval rw  100          --          100       1-60000
```

表 A-2 TCP 参数名变更

以前的 TCP 参数名	TCP 属性名
tcp_deferred_ack_interval	_deferred_ack_interval
tcp_local_dack_interval	_local_dack_interval
tcp_deferred_acks_max	_deferred_acks_max
tcp_local_dacks_max	_local_dacks_max

表 A-2 TCP 参数名变更 (续)

以前的TCP参数名	TCP属性名
tcp_wscale_always	_wscale_always
tcp_tstamp_always	_tstamp_always
tcp_xmit_hiwat	send_buf
tcp_recv_hiwat	recv_buf
tcp_max_buf	max_buf
tcp_cwnd_max	_cwnd_max
tcp_slow_start_initial	_slow_start_initial
tcp_slow_start_after_idle	_slow_start_after_idle
tcp_sack_permitted	sack
tcp_rev_src_routes	_rev_src_routes
tcp_time_wait_interval	_time_wait_interval
tcp_ecn_permitted	ecn
tcp_conn_req_max_q	_conn_req_max_q
tcp_conn_req_max_q0	_conn_req_max_q0
tcp_conn_req_min	_conn_req_min
tcp_rst_sent_rate_enabled	_rst_sent_rate_enabled
tcp_rst_sent_rate	_rst_sent_rate
tcp_keepalive_interval	_keepalive_interval
tcp_ip_abort_interval	_ip_abort_interval
tcp_rexmit_interval_initial	_rexmit_interval_initial
tcp_rexmit_interval_max	_rexmit_interval_max
tcp_rexmit_interval_min	_rexmit_interval_min
tcp_rexmit_interval_extra	_rexmit_interval_extra
tcp_tstamp_if_wscale	_tstamp_if_wscale
tcp_recv_hiwat_minmss	_recv_hiwat_minmss

UDP 参数名变更 (Oracle Solaris 11)

以下 UDP 参数已重命名为 UDP 属性。

可以使用类似于以下内容的语法设置 UDP 属性：

```
# ipadm set-prop -p send_buf=57344 udp
```

可以使用类似于以下内容的语法显示 UDP 属性信息：

```
# ipadm show-prop -p send_buf udp
PROTO PROPERTY          PERM CURRENT    PERSISTENT  DEFAULT  POSSIBLE
udp   send_buf            rw   57344         57344       57344    1024-2097152
```

表 A-3 UDP 参数名变更

以前的 UDP 参数名	UDP 属性名
udp_max_buf	max_buf
udp_xmit_hiwat	send_buf
udp_recv_hiwat	recv_buf

SCTP 参数名变更 (Oracle Solaris 11)

以下 SCTP 参数已重命名为 SCTP 属性。

可以使用类似于以下内容的语法设置 SCTP 属性：

```
# ipadm set-prop -p _max_init_retr=8 sctp
```

可以使用类似于以下内容的语法显示 SCTP 属性信息：

```
# ipadm show-prop -p _max_init_retr sctp
PROTO PROPERTY          PERM CURRENT    PERSISTENT  DEFAULT  POSSIBLE
sctp  _max_init_retr      rw   8             8           8        0-128
```

表 A-4 SCTP 参数名变更

以前的 SCTP 参数名	SCTP 属性名
sctp_max_init_retr	_max_init_retr
sctp_pa_max_retr	_pa_max_retr
sctp_pp_max_retr	_pp_max_retr
sctp_cwnd_max	_cwnd_max
sctp_ipv4_ttl	_ipv4_ttl

表 A-4 SCTP 参数名变更 (续)

以前的 SCTP 参数名	SCTP 属性名
sctp_heartbeat_interval	_heartbeat_interval
sctp_new_secret_interval	_new_secret_interval
sctp_initial_mtu	_initial_mtu
sctp_deferred_ack_interval	_deferred_ack_interval
sctp_ignore_path_mtu	_ignore_path_mtu
sctp_initial_ssthresh	_initial_ssthresh
sctp_ipv6_hoplimit	_ipv6_hoplimit
sctp_xmit_lowat	_xmit_lowat
sctp_xmit_hiwat	send_buf
sctp_rcv_hiwat	recv_buf
sctp_max_buf	max_buf
sctp_rto_min	_rto_min
sctp_rto_max	_rto_max
sctp_rto_initial	_rto_initial
sctp_cookie_life	_cookie_life
sctp_max_in_streams	_max_in_streams
sctp_initial_out_streams	_initial_out_streams
sctp_shutack_wait_bound	_shutack_wait_bound
sctp_maxburst	_maxburst
sctp_addip_enabled	_addip_enabled
sctp_prsctp_enabled	_prsctp_enabled

已过时或已删除的参数 (Oracle Solaris 11 和 Oracle Solaris 11.1)

下节介绍了最新 Oracle Solaris 发行版中已过时或已删除的参数。

NCA 参数

网络高速缓存加速器 (Network Cache Accelerator, NCA) 参数信息已过时并已删除。

consistent_coloring

自 Oracle Solaris 11.1 发行版起，此参数已过时。

rstchown

自 Oracle Solaris 11 发行版起，此参数已过时。

说明	指示 <code>chown</code> 系统调用的 POSIX 语义是否有效。POSIX 语义如下所示： <ul style="list-style-type: none"> ▪ 进程不能更改文件的所有者，除非进程以 <code>UID 0</code> 运行。 ▪ 进程不能将文件的组所有权更改为它目前不是其中成员的组，除非进程以 <code>UID 0</code> 运行。
	有关更多信息，请参见 chown(2) 。
数据类型	带符号整数
缺省值	1，表示使用 POSIX 语义
范围	0 = POSIX 语义未生效；1 = 使用 POSIX 语义
单位	切换（开/关）
动态?	是
验证	无
何时更改	当不需要 POSIX 语义时。请注意，关闭 POSIX 语义有可能造成各种安全漏洞。这样做还会产生以下可能性：某个用户将某个文件的所有权更改为另一个用户，从而使另一个用户能够在不受该用户或系统管理员干预的情况下得到该文件。
交付级别	过时

已过时的 TCP/IP 模块参数 (Oracle Solaris 11)

ip_multidata_outbound

此参数在 Oracle Solaris 11 发行版中已过时。

tcp_mdt_max_pbufs

此参数在 Oracle Solaris 11 发行版中已过时。

本手册的修订历史记录

本节介绍了本手册的修订历史记录。

- 第 167 页中的“当前版本：Oracle Solaris 11.1 发行版”
- 第 167 页中的“Oracle Solaris 发行版中新增或已更改的参数”

当前版本：Oracle Solaris 11.1 发行版

本手册的当前版本适用于 Oracle Solaris 11.1 发行版。

Oracle Solaris 发行版中新增或已更改的参数

以下各节介绍了新增、已更改或已过时的内核可调参数。

- Oracle Solaris 11.1：Oracle Solaris ZFS 可调参数信息位于第 3 章，[Oracle Solaris ZFS 可调参数](#)。
- Oracle Solaris 11.1：maxusers、max_nprocs、ngroups_max、pidmax 和 segkpsize 参数在 Oracle Solaris 11.1 发行版中已修改。有关更多信息，请参见第 2 章，[Oracle Solaris 内核可调参数](#)。
- Oracle Solaris 11.1：dnlc_dircache_percent 参数是 Oracle Solaris 11.1 发行版中的新增参数。有关更多信息，请参见第 59 页中的“[dnlc_dircache_percent](#)”。
- Oracle Solaris 11：rstchown 参数已过时。有关更多信息，请参见第 17 页中的“[Oracle Solaris 系统调优中有哪些新增内容?](#)”。
- Oracle Solaris 11：ipadm 命令替换了用于设置 TCP、IP、UDP 和 SCTP 属性的 ndd 命令。此外，网络参数的名称已更改，以便更好地关联到 ipadm 格式。有关更多信息，请参见第 113 页中的“[调优 IP 套件参数的概述](#)”。
- Oracle Solaris 11：本发行版包括 disp_rechoose_interval 参数。有关更多信息，请参见第 66 页中的“[disp_rechoose_interval](#)”。

- Oracle Solaris 11: 本发行版包括 `ngroups_max` 参数说明。有关更多信息, 请参见第 36 页中的“`ngroups_max`”。
- Oracle Solaris 11: 本发行版包括 `zfs_arc_min` 和 `zfs_arc_max` 参数说明。有关更多信息, 请参见第 74 页中的“`zfs_arc_min`”和第 74 页中的“`zfs_arc_max`”。
- Oracle Solaris 11: 本发行版包括多个 `igb` 和 `ixgbe` 网络驱动程序参数。有关更多信息, 请参见第 51 页中的“`igb` 参数”和第 52 页中的“`ixgbe` 参数”。
- Oracle Solaris 11: 本发行版包括了 `ddi_msix_alloc_limit` 参数, 它可用于增大设备实例可以分配的 MSI-X 中断次数。有关更多信息, 请参见第 51 页中的“`ddi_msix_alloc_limit`”。
- Oracle Solaris 11: 本发行版包括 `kmem_stackinfo` 参数, 启用该参数可监视内核线程栈使用情况。有关更多信息, 请参见第 49 页中的“`kmem_stackinfo`”。
- Oracle Solaris 11: 本发行版中提供了内存地址组参数。有关这些参数的更多信息, 请参见第 71 页中的“地址组参数”。
- Oracle Solaris 11: 参数信息已更新, 包括了 sun4v 系统。有关更多信息, 请参见下列参考信息:
 - 第 55 页中的“`maxphys`”
 - 第 60 页中的“`tmpfs:tmpfs_maxkmem`”
 - 第 68 页中的“特定于 SPARC 系统的参数”

索引

A

_addip_enabled, 147
_addrs_per_if, 117
autofs, 152
autoup, 31

C

_conn_req_max_q, 127
_conn_req_max_q0, 128
_conn_req_min, 128
_cookie_life, 145
cron, 152
_cwnd_max, 123,139

D

ddi_msix_alloc_limit 参数, 51
default_stksize, 26
default_tsb_size, 69
_deferred_ack_interval, 119,142
_deferred_acks_max, 120
desfree, 39
dhcpageant, 152
disp_rechoose_interval, 66,158
dnlc_dir_enable, 58
dnlc_dir_max_size, 59
dnlc_dir_min_size, 58
dnlc_dircache_percent, 59
doiflush, 32

dopageflush, 32

E

ecn, 127
enable_tsb_rss_sizing, 70

F

fastscan, 43
forwarding, 116
fs, 153
fsflush, 30
ftp, 153

H

handspreadpages, 44
_heartbeat_interval, 141
hires_tick, 68
hoplimit (ipv6), 116
hostmodel, 117

I

_icmp_err_burst, 114
_icmp_err_interval, 114
_icmp_return_data_bytes, 119
_ignore_path_mtu, 142

inetinit, 153
init, 153
_initial_mtu, 141
_initial_out_streams, 146
_initial_ssthresh, 142
intr_force, 52
intr_throttling, 53
_ip_abort_interval, 132
ip_squeue_fanout, 118
ip_squeue_worker_wait, 131
ipcl_conn_hash_size, 131
ipsec, 154
_ipv4_ttl, 140
_ipv6_hoplimit, 140

K

kbd, 154
_keepalive_interval, 132
keyserv, 154
kmem_flags, 48
kmem_stackinfo, 49

L

largest_anon_port, 130, 137, 148
lgrp_mem_pset_aware, 72
_local_dack_interval, 120
_local_dacks_max, 121
_local_slow_start_initial, 124
logevent_max_q_sz, 28
login, 154
lotsfree, 38
lpg_alloc_prefer, 71
lwp_default_stksize, 27

M

max_buf (SCTP), 144
max_buf (TCP), 123
max_buf (UDP), 136
_max_in_streams, 146

_max_init_retr, 138
max_nprocs, 35, 158
maxpgio, 45
maxphys, 55
maxpid, 35
maxuprc, 36
maxusers, 33, 158
min_percent_cpu, 44
minfree, 40
moddebug, 50
mpathd, 154
mr_enable, 51

N

ncsize, 57
ndd, 114
_new_secret_interval, 141
nfs_max_threads, 94
nfs:nacache, 105
nfs:nfs_allow_preepoch_time, 87
nfs:nfs_async_clusters, 102, 103
nfs:nfs_async_timeout, 104
nfs:nfs_cots_timeo, 87
nfs:nfs_disable_rddir_cache, 100
nfs:nfs_do_symlink_cache, 89
nfs:nfs_dynamic, 90
nfs:nfs_lookup_neg_cache, 91
nfs:nfs_nra, 95
nfs:nfs_shrinkreaddir, 98
nfs:nfs_write_error_interval, 99
nfs:nfs_write_error_to_cons_only, 100
nfs:nfs3_bsize, 101
nfs:nfs3_cots_timeo, 88
nfs:nfs3_do_symlink_cache, 89
nfs:nfs3_dynamic, 91
nfs:nfs3_jukebox_delay, 105
nfs:nfs3_lookup_neg_cache, 92
nfs:nfs3_max_threads, 94
nfs:nfs3_max_transfer_size, 106
nfs:nfs3_max_transfer_size_clts, 107
nfs:nfs3_max_transfer_size_cots, 108
nfs:nfs3_nra, 96
nfs:nfs3_pathconf_disable_cache, 86

nfs:nfs3_shrinkreaddir, 99
 nfs:nfs4_async_clusters, 104
 nfs:nfs4_bsize, 101
 nfs:nfs4_cots_timeo, 88
 nfs:nfs4_do_symlink_cache, 90
 nfs:nfs4_lookup_neg_cache, 93
 nfs:nfs4_max_threads, 95
 nfs:nfs4_max_transfer_size, 107
 nfs:nfs4_nra, 96
 nfs:nfs4_pathconf_disable_cache, 86
 nfs:nrnode, 97
 nfslogd, 155
 ngroups_max, 36
 noexec_user_stack, 29
 nss, 155
 nstrpush, 64

O

Oracle 数据库调优, ZFS 文件系统, 81

P

pageout_reserve, 41
 pages_before_pager, 45
 pages_pp_maximum, 42
 passwd, 155
 _pathmtu_interval, 118
 physmem, 26
 pidmax, 34, 158
 _policy_mask, 138
 _pp_max_retr, 139
 primarycache, ZFS 文件系统属性, 81
 _prsctp_enabled, 147
 pt_cnt, 62
 pt_max_pty, 63
 pt_pctofmem, 63

R

recordsize, ZFS 文件系统属性, 80
 recv_buf (SCTP), 143

recv_buf (TCP), 123
 recv_buf (UDP), 136
 _recv_hiwat_minmss, 135
 reserved_procs, 34
 _respond_to_echo_broadcast, 115
 _respond_to_echo_multicast, 115
 _rev_src_routes, 126
 _rexit_interval_extra, 134
 _rexit_interval_initial, 133
 _rexit_interval_max, 133
 _rexit_interval_min, 134
 rlim_fd_cur, 57
 rlim_fd_max, 56
 rpcmod:clnt_idle_timeout, 109
 rpcmod:clnt_max_conns, 108
 rpcmod:cotsmaxdupreqs, 111
 rpcmod:maxdupreqs, 110
 rpcmod:svc_default_stksize, 110
 rpcmod:svc_idle_timeout, 109
 _rst_sent_rate, 129
 _rst_sent_rate_enabled, 129
 rstchown, 165
 _rto_max, 145
 _rto_min, 144
 rx_copy_threshold, 55
 rx_limit_per_intr, 53
 rx_queue_number, 53
 rx_ring_size, 54

S

sack, 125
 sctp_maxburst, 147
 secondarycache, ZFS 文件系统属性, 81
 segkpsize, 158
 segspt_minfree, 66
 send_buf (SCTP), 143
 send_buf (TCP), 122
 send_buf (UDP), 135
 _send_redirects, 115
 _shutack_wait_bound, 146
 _slow_start_after_idle, 125
 _slow_start_initial, 124
 slowscan, 43

smallest_anon_port, 130,137,148
strmsgsz, 64,65
su, 155
swapfs_minfree, 47,159
swapfs_reserve, 46
syslog, 155

T

tar, 155
throttlefree, 40
_time_wait_interval, 126
timer_max, 68
tmpfs_maxkmem, 60
tmpfs_minfree, 61
tsb_alloc_hiwater, 69
tsb_rss_factor, 70
_tstamp_always, 122
_tstamp_if_wscale, 135
ttl (ipv4), 116
tune_t_fsflushr, 30
tune_t_minarmem, 42
tx_copy_threshold, 55
tx_queue_number, 52
tx_ring_size, 54

U

utmpd, 156

W

_wscale_always, 121

X

_xmit_lowat, 143

Z

zfs_arc_max, 74,157
zfs_arc_min, 74,157
zfs_prefetch_disable, 75
ZFS 文件系统, 调优 Oracle 数据库, 81
ZFS 文件系统属性
 primarycache, 81
 recordsize, 80
 secondarycache, 81