在 Oracle® Solaris 11.2 中管理审计

文件号码 E53976
2014 年 7 月

ORACLE

目录

使用此文档

《在 *Oracle*® *Solaris 11.2* 中管理审计》介绍了 Oracle Solaris 的审计功能。

- **概述** – 介绍了如何在 Oracle Solaris 系统或系统网络上管理审计。
- **目标读者** – 负责实现企业网络安全的系统管理员。
- **必备知识** – 熟悉安全概念和术语。

产品文档库

有关本产品的最新信息和已知问题均包含在文档库中，网址为：http://www.oracle.com/pls/topic/lookup?ctx=E56344。

获得 Oracle 支持

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

反馈

可以在 http://www.oracle.com/goto/docfeedback 上提供有关此文档的反馈。

♦ ♦ ♦ 第 1 章

1 关于 Oracle Solaris 中的审计

Oracle Solaris 的审计子系统保留了有关如何使用系统的记录。审计服务包括帮助分析审计数据的工具。

本章介绍 Oracle Solaris 中审计的工作原理:

有关规划建议，请参见第 2 章 规划审计。有关在站点上配置审计的过程，请参见以下各章：

有关参考信息，请参见第 7 章 审计参考。

Oracle Solaris 中审计服务的新增功能

本节为现有客户重点介绍了有关 Oracle Solaris 审计服务中重要新功能的信息。

- 来自 Oracle Solaris 系统的审计记录可以插入到 Oracle Audit Vault and Database Firewall，然后可以使用这些记录来获得关于 Oracle Solaris 系统审计事件的信息。
- 审计配置文件（`audit_class`、`audit_event` 和 `audit_warn`）设置有两个软件包属性。`preserve=renamenew` 属性允许修改文件，并且修改将保留在软件包更新和软件包修复中。`overlay=allow` 属性允许将文件替换为客户创建的软件包中的文件。

什么是审计？

审计是指收集有关系统资源使用情况的数据。审计数据提供安全相关的系统事件的记录。以后便可以使用此数据来指定主机上执行的操作的职责。

成功的审计应包括两个安全功能：识别和验证。每次登录时，用户提供用户名并成功通过 PAM（Pluggable Authentication Module，可插拔验证模块）验证后，将生成唯一且保持不变的审计用户 *ID* 并与该用户相关联，还会生成唯一的审计会话 ID 并与该用户的进程相关联。在该登录会话期间启动的每个进程都会继承此审计会话 ID。从一个用户切换为另一个用户时，系统使用同一审计用户 ID 跟踪所有用户操作。有关切换身份的更多详细信息，请参见 su(1M) 手册页。请注意，缺省情况下，某些操作（例如引导和关闭系统）始终都需要进行审计。

通过审计服务可以实现以下操作：

- 监视主机上发生的与安全相关的事件
- 在网络范围的审计迹中记录事件
- 检测误用或未经授权的活动
- 查看访问模式以及个人和对象的访问历史记录
- 发现绕过保护机制的尝试
- 发现用户更改身份时对特权的扩展使用

注 - 为了维护安全性，并不是所有的审计事件（如更改口令）都是可见的。有关更多详细信息，请参见“审计记录和审计标记” [14]。

审计术语和概念

以下术语用于说明审计服务。有些定义包含指向更完整说明的链接。

audit class（审计类）	一组审计事件。审计类提供选择一组要审计的事件的方法。 有关更多信息，请参见“审计类和预选” [13]以及 audit_flags(5)、audit_class(4) 和 audit_event(4) 手册页。
audit file system（审计文件系统）	二进制格式的审计文件系统信息库。 有关更多信息，请参见“审计日志” [15]和 audit.log(4) 手册页。
audit event（审计事件）	可审计的与安全相关的系统操作。为了便于选择，将事件分为多个审计类。 有关更多信息，请参见“审计事件” [12]和 audit_event(4) 手册页。

audit flag（审计标志）

作为参数提供给命令或关键字的审计类。标志的前缀可以是加号或减号，+ 表示对类的成功事件进行审计，而 - 表示对类的失败事件进行审计。在加号前面加上插入记号 ^ 表示将不审计成功事件 (^+)，而在减号前面加上该插入记号表示将不审计失败事件 (^-)。

有关更多信息，请参见 audit_flags(5) 手册页和“审计类语法” [105]。

audit plugin（审计插件）

用于将队列中的审计记录传输到指定位置的模块。audit_binfile 插件可创建二进制审计文件。而这些二进制文件组成了审计迹，存储在审计文件系统中。audit_remote 插件可将二进制审计记录发送到远程系统信息库。audit_syslog 插件可将选定的审计记录汇总到 syslog 日志中。

有关更多信息，请参见“审计插件模块” [14] 以及模块手册页 audit_binfile (5)、audit_remote(5) 和 audit_syslog(5)。

audit policy（审计策略）

一组可以在您的站点中启用或禁用的审计选项。您可以指定是否记录某些类型的审计数据，以及在审计队列变满时是否要暂停可审计的操作。

有关更多信息，请参见“了解审计策略” [29]和 auditconfig(1M) 手册页。

audit record（审计记录）

审计队列中收集的审计数据。一条审计记录描述一个审计事件。每条审计记录由多个审计标记组成。

有关更多信息，请参见“审计记录和审计标记” [14]和 audit.log(4) 手册页。

audit token（审计标记）

审计记录或审计事件字段。每个审计标记描述审计事件的一个属性，例如用户、组、程序或其他对象。

有关更多信息，请参见“审计标记格式” [110]和 audit.log(4) 手册页。

audit trail（审计迹）

一个或多个审计文件的集合，用于存储使用缺省插件 audit_binfile 的所有被审计系统上的审计数据。

有关更多信息，请参见“审计迹” [108]。

local auditing（本地审计）

收集在本地系统上生成的审计记录。这些记录可以是在全局区域或非全局区域（或者两者）中生成的。

有关更多信息，请参见“审计插件模块” [14]。

post-selection（后选）

有关要在审计迹中检查哪些审计事件的选择。缺省活动插件 audit_binfile 可创建审计迹。后选工具 auditreduce 命令将从审计迹中选择记录。

有关更多信息，请参见 auditreduce(1M) 和 praudit(1M) 手册页。

preselection（预选）

有关要监视哪些审计类的选择。将在审计队列中收集预选的审计类的审计事件。由于不会审计未预选的审计类，因此这些审计类的事件将不会出现在队列中。

有关更多信息，请参见“审计类和预选” [13]和 audit_flags(5) 和 auditconfig(1M) 手册页。

public object（公共对象）

由 root 用户拥有且任何人都可读取的文件。例如，/etc 目录和 /usr/bin 目录中的文件就是公共对象。不会审计只读事件的公共对象。例如，即使预选了 file_read (fr) 审计类，也不会审计公共对象的读取。您可以通过更改 public 审计策略选项来覆盖缺省值。

remote auditing（远程审计）

审计远程服务器 (Audit Remote Server, ARS) 接收并存储来自正被审计且配置有活动 audit_remote 插件的系统的审计记录。为了区分被审计系统与 ARS，可以将被审计系统称为“本地被审计系统”。

有关更多信息，请参见 auditconfig(1M) 手册页中的 -setremote 选项和“审计远程服务器” [106]。

审计事件

审计事件代表某系统上可审计的操作。审计事件在 /etc/security/audit_event 文件中列出。每个审计事件均连接到一个系统调用或用户命令，并指定给一个或多个审计类。有关 audit_event 文件的格式说明，请参见 audit_event(4) 手册页。

例如，AUE_EXECVE 审计事件将审计 execve() 系统调用。命令 auditrecord -e execve 显示以下条目：

```
# auditrecord -e execve
execve
system call execve              See execve(2)
event ID    23                  AUE_EXECVE
class       ps,ex               (0x0000000040100000)
header
path
[attribute]                 omitted on error
[exec_arguments]            output if argv policy is set
[exec_environment]          output if arge policy is set
subject
[use_of_privilege]
return
```

如果预选审计类 ps 或 ex，则每个 execve() 系统调用都将记录在审计队列中。

审计过程处理可归属和无归属事件。审计策略将事件分为同步事件和异步事件，如下所示：

- **可归属事件** – 可归属到某个用户的事件。execve（） 系统调用可归属到某个用户，因此将该调用视为可归属事件。所有的可归属事件都是同步事件。
- **无归属事件** – 在内核中断级别发生的事件，或在验证用户之前发生的事件。na 审计类处理无归属审计事件。例如，引导系统便是一个无归属事件。多数无归属事件都是异步事件。但是，具有关联进程的无归属事件（如登录失败）是同步事件。
- **同步事件** – 与系统中的进程关联的事件。同步事件占系统事件的大多数。
- **异步事件** – 与任何进程无关联的事件，因此既不会阻止进程，随后也不会启动进程。例如，初始系统引导和 PROM 进入和退出事件都是异步事件。

除了审计服务定义的审计事件，第三方应用程序也可以生成审计事件。审计事件编号 32768 到 65535 适用于第三方应用程序。供应商需要联系其 Oracle Solaris 代表，以保留事件编号并获取对审计界面的访问权限。

审计类和预选

每个审计事件属于一个审计类。审计类是用于容纳大量审计事件的方便容器。预选要审计的类时，该类中的所有事件都将记录在审计队列中。例如，预选 ps 审计类时，将记录 execve（）、fork（） 以及其他系统调用。

可以预选系统中的事件和特定用户启动的事件。

- **系统范围的预选** – 通过在 auditconfig 命令中使用 -setflags 和 -setnaflags 选项指定系统范围内的审计缺省值。

注 - 如果已设置 perzone 策略，则可以在每个区域中指定缺省审计类。对于 perzone 审计，缺省值为区域范围，而非系统范围。

- **特定于用户的预选** – 通过为用户配置审计标志，指定各个用户的审计值与系统范围的审计缺省值之间的差异。useradd、roleadd、usermod 以及 rolemod 命令将 audit_flags 安全属性放置在 user_attr 数据库中。profiles 命令将权限配置文件的审计标志放置在 prof_attr 数据库中。

  审计预选掩码确定要针对用户审计的事件类。有关用户预选掩码的说明，请参见“进程审计特征” [108]。有关所使用的配置审计标志，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“所指定权限的搜索顺序”。

审计类在 /etc/security/audit_class 文件中定义。每个项都包含类的审计掩码、类的名称，以及类的描述性名称。例如，lo 和 ps 类定义在 audit_class 文件中显示为：

```
0x0000000000001000:lo:login or logout
0x0000000000100000:ps:process start/stop
```

审计类包括以下两个全局类：all 和 no。audit_class(4) 手册页中介绍了这些审计类。有关类的列表，请阅读 /etc/security/audit_class 文件。

可以配置审计事件到类的映射。可以从类中删除事件、向类中添加事件，还可以创建新类以包含选定事件。有关过程，请参见如何更改审计事件的类成员身份 [49]。要查看映射到类的事件，请使用 auditrecord -c *class* 命令。

审计记录和审计标记

每条审计记录都会记录一个发生的审计事件。该记录包含操作执行者、受影响的文件、尝试执行的操作以及操作发生的时间和位置等信息。以下示例显示了带有三个标记（header、subject 和 return）的 login 审计记录：

```
header,69,2,login - local,,example_system,2010-10-10 10:10:10.020 -07:00
subject,jdoe,jdoe,staff,jdoe,staff,1210,4076076536,69 2 example_system
return,success,0
```

为每个审计事件保存的信息类型由一组审计标记进行定义。每次为事件创建审计记录时，记录中都会包含为该事件定义的部分或全部标记。事件的性质决定了要记录的标记。在前面的示例中，每行都以审计标记的名称开头。标记名称后跟审计标记的内容。header、subject 和 return 审计标记一起组成了 login - local 审计记录。要显示组成审计记录的标记，请使用 auditrecord -e *event* 命令。

注 - 具有 sensitive 系统属性的文件未将其内容或内容更改包含在审计记录中。该属性可确保任何人都无法访问特定文件中的口令、PIN、密钥等敏感信息。有关更多详细信息，请参阅 pfedit(1M) 手册页。

有关每个审计标记结构的详细说明和 praudit 输出示例，请参见“审计标记格式” [110]。有关审计标记的二进制流的说明，请参见 audit.log(4) 手册页。

审计插件模块

审计插件模块将审计记录从审计队列定向到文件或系统信息库。至少有一个插件必须处于活动状态。缺省情况下，audit_binfile 插件处于活动状态。可使用 auditconfig -setplugin *plugin-name* 命令配置插件。

审计服务提供以下插件：

- audit_binfile 插件 – 处理将审计队列发送到二进制审计文件的过程。有关更多信息，请参见 audit.log(4) 手册页。
- audit_remote 插件 – 处理将二进制审计记录从审计队列安全发送到已配置的远程服务器的过程。audit_remote 插件使用 libgss（） 库验证服务器。出于保密性和完整性的考虑，传输过程受到保护。

- audit_syslog 插件 – 处理将选定记录从审计队列发送到 syslog 日志的过程。

有关如何配置插件的信息，请参见 auditconfig(1M) 手册页。有关插件配置的示例，请参见“配置审计日志” [65]中的任务。有关插件的信息，请参见 audit_binfile (5)、audit_remote(5) 和 audit_syslog(5) 手册页。

审计日志

审计记录是在审计日志中收集的。对于审计记录，审计服务提供三种输出模式。

- 称为审计文件的日志以二进制格式存储审计记录。系统或站点的审计文件集提供完整的审计记录。完整的审计记录称为审计迹。这些日志由 audit_binfile 插件创建，可通过 praudit 和 auditreduce 后选命令查看。
- audit_remote 插件可将审计记录流化处理到远程系统信息库。该系统信息库负责维护审计迹并提供后选工具。
- syslog 实用程序收集并存储审计记录的文本摘要。syslog 记录不是完整的记录。以下示例显示了 login 审计记录的 syslog 项：

```
Oct 10  10:10:20 example_system auditd: [ID 6472 audit.notice] \
login - login ok session 4076172534 by root as root:other
```

站点可以配置审计以收集所有格式的审计记录。您可以对您的站点中的系统进行配置，以在本地使用二进制模式，将二进制文件发送到远程系统信息库，以及使用 syslog 模式。下表对二进制审计记录和 syslog 审计记录进行了比较。

表 1-1 二进制审计记录、远程审计记录以及 syslog 审计记录的比较

功能	二进制和远程记录	syslog 记录
协议	二进制 – 写入文件系统	将 UDP 用于远程日志记录
	远程 – 流化处理到远程系统信息库	
数据类型	二进制	文本
记录长度	无限制	每条审计记录最多 1024 个字符
位置	二进制 – 存储在系统上的 zpool 中	存储在 syslog.conf 文件中指定的位置
	远程 – 远程系统信息库	
配置方式	二进制 – 在 audit_binfile 插件上设置 p_dir 属性	使 audit_syslog 插件处于活动状态并配置 syslog.conf 文件
	远程 – 在 audit_remote 插件上设置 p_hosts 属性，并使该插件处于活动状态	
读取方式	二进制 – 在批处理模式下，浏览器通常以 XML 格式输出	实时读取，或者通过为 syslog 创建的脚本进行搜索
	远程 – 系统信息库指定相应过程	纯文本输出
完整性	保证完整，并且以正确的顺序显示	不能保证完整

功能	二进制和远程记录	syslog 记录
时间戳	国际协调时间 (Coordinated Universal Time, UTC)	审计的系统时间

有关插件和审计日志的更多信息，请参阅以下内容：

- audit_binfile (5) 手册页
- audit_syslog(5) 手册页
- audit.log(4) 手册页
- 如何为审计迹指定审计空间 [69]
- 如何配置 syslog 审计日志 [78]

关于二进制记录

二进制记录提供最高的安全性和最大的覆盖范围。二进制输出满足安全证书的要求，例如通用准则 (http://www.commoncriteriaportal.org/) 审计要求。

audit_binfile 插件可将记录写入对其进行保护以防止偷窥的文件系统中。在单个系统上，将收集所有二进制记录并按顺序显示它们。当某个审计迹内的系统分布于不同的时区时，可以参考二进制日志中的 UTC 时间戳进行精确比较。使用 praudit -x 命令可在浏览器中查看 XML 格式的记录。还可以使用脚本来解析 XML 输出。

audit_remote 插件可将记录写入远程系统信息库。该系统信息库负责存储和后选。

关于 syslog 审计记录

相反，syslog 记录可能会提供更大的便利性和灵活性。例如，您可以从各种源收集 syslog 数据。此外，当您监视 syslog.conf 文件中的 audit.notice 事件时，syslog 实用程序会记录一条带有当前时间戳的审计记录摘要。您可以使用为来自各种源（包括工作站、服务器、防火墙和路由器）的 syslog 消息开发的同一管理和分析工具。可以实时查看记录，并将其存储在远程系统中。

通过使用 syslog.conf 远程存储审计记录，可以保护日志数据免遭攻击者改动或删除。不过，请注意 syslog 模式存在以下缺点。

- 这些记录容易遭受拒绝服务、伪装源地址等网络攻击。
- UDP 协议会丢包或无序发送包。
- syslog 项的限制为 1024 个字符，因此可能会截断日志中的某些审计记录。
- 在单个系统上，可能并非所有的审计记录都会收集，而且收集到的记录可能不会按顺序显示。
- 每个审计记录会加盖本地系统的日期和时间。因此，您不能依靠时间戳为几个系统构建审计迹。

存储和管理审计迹

audit_binfile 插件处于活动状态时，审计文件系统以二进制格式保留审计文件。典型的安装使用 /var/audit 文件系统，也可以使用其他文件系统。所有审计文件系统的内容组成了审计迹。审计记录按以下顺序存储在这些文件系统中：

- **主审计文件系统** – /var/audit 文件系统，即用于系统审计文件的缺省文件系统
- **辅助审计文件系统** – 按管理员意愿在其中放置系统审计文件的文件系统

这些文件系统作为 audit_binfile 插件的 p_dir 属性参数来指定。对于某个文件系统，列表中位于此文件系统前面的文件系统已满时才会使用此文件系统。有关包含文件系统项列表的示例，请参见如何为审计文件创建 ZFS 文件系统 [66]。

将审计文件放置在缺省审计根目录下可帮助审计审阅者审阅审计迹。auditreduce 命令使用审计根目录来查找审计迹中的所有文件。缺省审计根目录为 /var/audit。

可以使用带有以下选项的 auditreduce 命令：

- auditreduce 命令的 -M 选项可用于指定来自特定计算机的审计文件。
- -S 选项可用来指定不同的审计文件系统。

有关 auditreduce 命令的使用示例，请参见如何合并审计迹中的审计文件 [89]。有关更多信息，请参见 auditreduce(1M) 手册页。

审计服务提供用于合并和过滤审计迹文件的命令。auditreduce 命令可以合并审计迹中的审计文件。此命令还可以过滤文件以查找特定事件。praudit 命令读取二进制文件。praudit 命令的选项提供适合借助脚本和浏览器显示的输出。

确保时间戳可靠

合并多个系统中的审计日志时，这些系统上的日期和时间必须准确。同样，将审计日志发送到远程系统时，记录系统和系统信息库系统必须具有准确的时钟。网络时间协议 (Network Time Protocol, NTP) 可以使各个系统时钟保持准确和相互协调。有关更多信息，请参见《Oracle Solaris 11.2 网络服务介绍 》中的第 3 章 “与时间相关的服务”和 xntpd (1M) 手册页。

管理远程系统信息库

配置 audit_remote 插件后，审计记录将由远程系统信息库接收。ARS 为审计记录提供了接收者。审计记录通过受保护的连接以流方式传输到 ARS，并且能够像在本地存储时那样存储它们。要配置 ARS，请参见如何配置审计文件的远程系统信息库 [74]。有关 ARS 的描述，请参见“审计远程服务器” [106]和 ars(5) 手册页。

审计如何与安全相关？

审计通过显示可疑或异常的系统使用模式来帮助检测潜在的安全违规。审计还提供一种根据可疑操作追溯到特定用户的方法，因此可以起到一种威慑的作用。知道自己的活动正在被审计的用户很少会尝试执行恶意操作。

要保护计算机系统，特别是网络中的系统，需要在系统进程或用户进程开始之前具备控制活动的机制。安全性要求系统上安装有在活动发生时用于监视活动的工具，同时还要求在活动发生后报告活动。

在用户登录或开始系统进程之前设置审计参数，因为大部分审计活动都涉及监视当前事件和报告符合指定参数的事件。第 2 章 规划审计和第 3 章 管理审计服务中详细介绍了审计服务如何监视和报告这些事件。

审计不能防止黑客未经授权的侵入。但是，审计服务可以报告特定用户在特定日期和时间执行了特定操作之类的信息。审计报告可以按登录路径和用户名来标识用户。此类信息可立即报告给终端和文件，以供以后分析。因此，审计服务提供的数据有助于确定以下内容：

- 系统安全如何受到威胁
- 需要关闭哪些漏洞来确保期望的安全级别

审计工作原理

在发生指定事件时，审计会生成审计记录。通常情况下，生成审计记录的事件包括：

- 系统启动和系统关闭
- 登录和注销
- 进程创建或进程销毁，或线程创建或线程销毁
- 打开、关闭、创建、销毁或重命名对象
- 使用权限
- 识别操作和验证操作
- 由进程或用户执行的权限更改
- 管理操作，例如安装软件包
- 特定于站点的应用程序

审计记录从以下三个源生成：

- 应用程序
- asynchronous audit event（异步审计事件）的结果
- 进程系统调用的结果

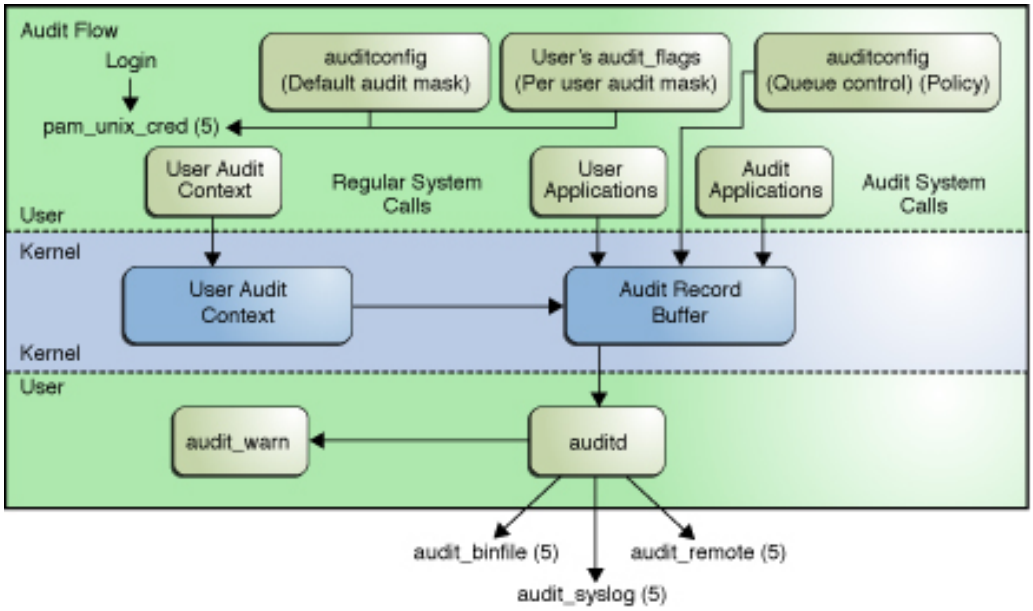
捕获相关的事件信息后，会将这些信息格式化为审计记录。每条审计记录都包含识别事件的信息、导致事件的原因、事件发生的时间，以及其他相关信息。该记录随后会被放

置在审计队列中并发送到活动插件进行存储。尽管所有插件都可以处于活动状态，但至少有一个插件必须处于活动状态。“配置审计的方式” [19]和“审计插件模块” [14]中介绍了这些插件。

配置审计的方式

在系统配置期间，可以预选要监视的审计记录的类。还可以针对单个用户微调执行审计的程度。下图显示了 Oracle Solaris 审计流程的详细信息。

图 1-1 审计流程



在内核中收集审计数据后，插件会将数据分发到相应位置。

- `audit_binfile` 插件可将二进制审计记录放置在 `/var/audit` 文件系统中。缺省情况下，`audit_binfile` 插件处于活动状态。通过后选工具，可以检查审计迹中感兴趣的部分。

  审计文件可以存储在一个或多个 ZFS 池中。这些池可以位于不同的系统中，也可以位于不同但链接的网络中。相互链接的审计文件集合称为审计迹。
- `audit_remote` 插件可将受保护链接中的二进制审计记录发送到远程系统信息库。
- `audit_syslog` 插件可将审计记录的文本摘要发送到 `syslog` 实用程序。

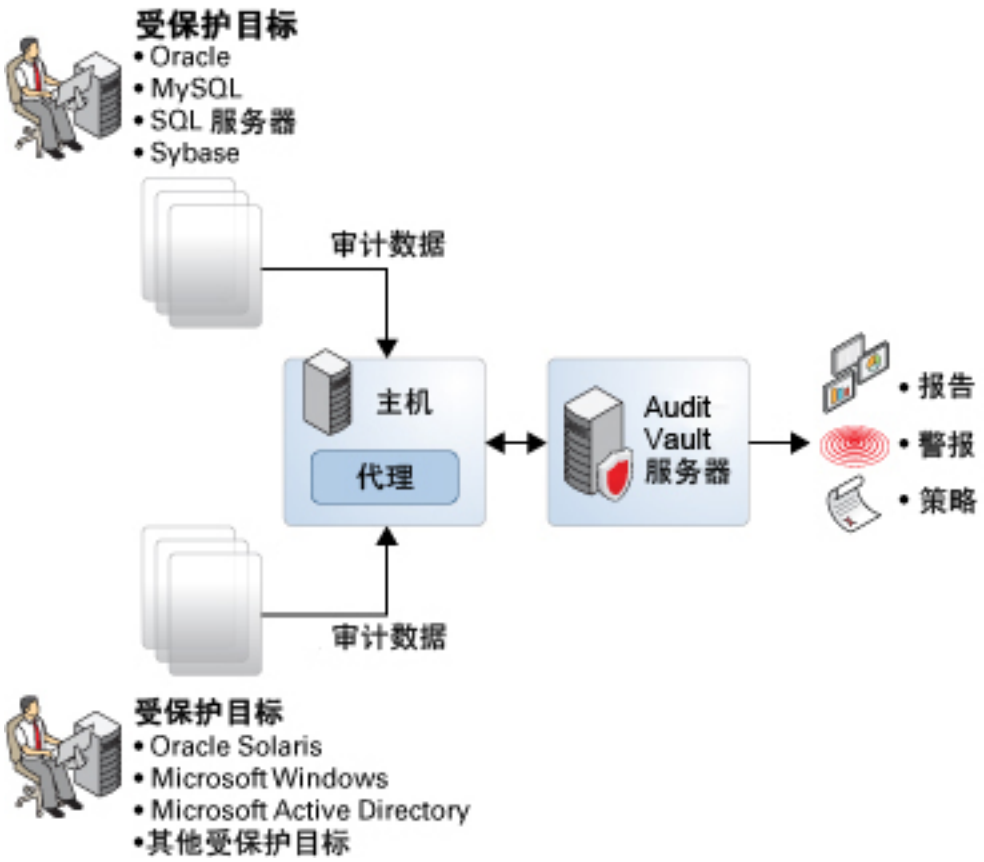
安装非全局区域的系统可以从全局区域以相同的方式审计所有区域。还可以配置这些系统，使其收集非全局区域中的不同记录。有关更多信息，请参见“审计和 Oracle Solaris 区域” [104]。

将 Oracle Audit Vault and Database Firewall 用于存储和分析审计记录

来自 Oracle Solaris 系统的审计记录可以插入到 Oracle Audit Vault and Database Firewall 发行版 12.1.0.0。Oracle Audit Vault and Database Firewall 自动合并和监视来自 Oracle 和非 Oracle 数据库的审计数据。然后，可以使用 Oracle Audit Vault and Database Firewall 分析和报告 Oracle Solaris 系统的审计事件。有关更多信息，请参见 Oracle Audit Vault and Database Firewall (http://www.oracle.com/technetwork/products/audit-vault/overview/index.html)。

下图显示了 Oracle Audit Vault and Database Firewall 如何从指定的受保护目标收集 Oracle Solaris 审计记录。受保护目标是存储审计记录或数据的任何系统。

图 1-2 Oracle Solaris and Audit Vault



指定在主机上运行与 Oracle Audit Vault and Database Firewall 进行通信的 AV 代理。代理使得 Oracle Audit Vault and Database Firewall 从受保护目标接收审计数据并对其进行处理。代理从受保护目标上的指定审计迹读取审计记录。这些审计记录采用本机二进制格式进行编码。代理将数据转换为 Oracle Audit Vault and Database Firewall 可解析的格式。Oracle Audit Vault and Database Firewall 接收数据并根据需要为管理员和安全管理器生成报告。

代理可以安装在受保护目标（而不是单独的主机或系统）上。还可以配置带有代理的多台主机，以连接到 Audit Vault 服务器。然而，注册受保护目标时，请指定 AV 服务器与之通信以获取审计数据的特定主机。

要将 Oracle Audit Vault and Database Firewall 配置为接受来自 Oracle Solaris 受保护目标和非 Oracle Solaris 受保护目标的审计记录，请确保已在指定的主机系统上安装和启动了代理。有关更多信息，请参见 Oracle Audit Vault and Database

Firewall Documentation (http://www.oracle.com/technetwork/products/audit-vault/documentation/index.html)（Oracle Audit Vault and Database Firewall 文档）。

装有 Oracle Solaris 区域的系统上的审计

区域是在 Oracle Solaris OS 单一实例中创建的虚拟化操作系统环境。审计服务审计整个系统，包括区域中的活动。安装了非全局区域的系统可以运行单一审计服务，以相同方式审计所有区域。它还可为每个区域（包括全局区域）运行一个审计服务。

满足以下条件的站点可以运行单一审计服务：

- 站点需要单映像审计迹。
- 非全局区域用作应用程序容器。区域是一个管理域的组成部分。也就是说，非全局区域都没有定制的命名服务文件。

  如果系统上的所有区域都在一个管理域中，可以使用 `zonename` 审计策略区分在不同区域配置的审计事件。
- 管理员希望降低审计开销。全局区域管理员以同样的方式审计所有区域。而且，全局区域的审计守护进程为系统上的所有区域服务。

满足以下条件的站点可为每个区域运行一个审计服务：

- 站点不需要单映像审计迹。
- 非全局区域有定制的命名服务文件。这些独立的管理域通常充当服务器。
- 各区域管理员希望控制他们管理的区域中的审计。在按区域审计中，区域管理员可以决定是在他们管理的区域中启用还是禁用审计。

按区域审计的优点在于，可为每个区域定制审计迹，并且能够按区域禁用审计。但是，这些优点可能会被管理开销抵销。每个区域管理员都必须管理审计。每个区域运行自己的审计守护进程，有自己的审计队列和审计日志。必须管理这些审计日志。

♦ ♦ ♦ 第 2 章

2 规划审计

本章介绍了如何为您的 Oracle Solaris 安装规划审计服务的定制：

- “规划审计的概念” [23]
- “规划审计” [25]
- “了解审计策略” [29]
- “控制审计成本” [31]
- “有效审计” [32]

有关审计的概述，请参见第 1 章 关于 Oracle Solaris 中的审计。有关在站点上配置审计的过程，请参见以下各章：

- 第 3 章 管理审计服务
- 第 4 章 监视系统活动
- 第 5 章 使用审计数据
- 第 6 章 分析和解决审计服务问题

有关参考信息，请参见第 7 章 审计参考。

规划审计的概念

您需要认真选择要审计的活动类型，同时还需要收集有用的审计信息。您还需要仔细规划要审计的对象及内容。如果使用缺省 `audit_binfile` 插件，审计文件会快速增长进而填满可用空间，因此必须分配足够的磁盘空间。

规划单一系统审计迹

注 - 只适用于对 `audit_binfile` 插件实现单一系统审计迹。

单一管理域中的系统可以创建单系统映像审计迹。

要为站点创建单一系统映像审计迹，请遵循以下要求：

- 针对所有系统，使用相同的命名服务。

要正确解释审计记录，passwd、group 和 hosts 文件必须一致。

- 在所有系统上以相同的方式配置审计服务。有关显示和修改服务设置的信息，请参见 auditconfig(1M) 手册页。
- 对于所有系统，使用相同的 audit_warn、audit_event 和 audit_class 文件。

有关对系统启用审计的其他注意事项，请参阅如何规划要审计的对象及内容 [25]。

规划区域中的审计

如果系统包含非全局区域，可通过审计全局区域来审计这些区域，或者可以单独为每个非全局区域配置、启用和禁用审计服务。例如，您可以仅审计非全局区域，而不审计全局区域。

有关如何权衡选择的介绍，请参见“装有 Oracle Solaris 区域的系统上的审计” [22]。

在区域中实现审计时，可使用以下选项。

对所有区域实现一个审计服务

以相同方式审计所有区域时，可创建单映像审计迹。使用 audit_binfile 或 audit_remote 插件且系统上的所有区域都属于一个管理域时，会出现单映像审计迹。然后，可以很容易地比较审计记录，因为每个区域的记录都以相同的设置预选。

该配置将所有区域视为一个系统的组成部分。全局区域在一个系统上只运行一个审计服务，并收集所有区域的审计记录。您只能在全局区域中定制 audit_class 和 audit_event 文件，然后将这些文件复制到各个非全局区域。

对所有区域配置一个审计服务时，请使用以下指导原则。

- 对每个区域使用相同的命名服务。

注 - 如果已在非全局区域中定制命名服务文件，但未设置 perzone 策略，那么需要谨慎地使用审计工具来选择可用的记录。一个区域中的用户 ID 可以指不同区域中使用相同 ID 的不同用户。

- 启用包括区域名称在内的审计记录。

  要使区域名称成为审计记录的一部分，请在全局区域中设置 zonename 策略。然后，可以使用 auditreduce 命令按区域从审计迹中选择审计事件。有关示例，请参见 auditreduce(1M) 手册页。

要规划单映像审计迹，请参阅如何规划要审计的对象及内容 [25]。从第一步开始。全局区域管理员还必须留出部分存储，如如何规划审计记录的磁盘空间 [27]中所述。

每个区域实现一个审计服务

如果不同区域使用不同的命名服务数据库，或者区域管理员想要控制他们区域中的审计，请选择配置按区域审计。

注 - 要审计非全局区域，必须设置 perzone 策略，但不必在全局区域中启用审计服务。配置非全局区域审计，并从全局区域单独启用和禁用其审计服务。

- 配置按区域审计时，需要在全局区域中设置 perzone 审计策略。如果在第一次引导非全局区域之前设置按区域审计，则在第一次引导区域时开始审计。要设置审计策略，请参见如何配置每区域审计 [61]。
- 每个区域管理员为该区域配置审计。

  非全局区域管理员可以设置 perzone 和 ahlt 以外的全部策略选项。
- 每个区域管理员都可以启用或禁用区域中的审计。
- 要在审查期间生成可追溯到其来源区域的记录，请设置 zonename 审计策略。

注 - 按区域审计时，如果 audit_binfile 插件处于活动状态，每个区域管理员还必须为每个区域留出部分存储，如如何规划审计记录的磁盘空间 [27]中所述。有关其他的规划说明，请参见如何规划要审计的对象及内容 [25]。

规划审计

以下任务列表列出了规划磁盘空间以及要记录的事件时所需执行的主要任务。

表 2-1 规划审计任务列表

任务	参考
确定要审计的对象及内容	如何规划要审计的对象及内容 [25]
规划审计迹的存储空间	如何规划审计记录的磁盘空间 [27]
规划向远程服务器传输审计迹	以流方式将审计记录传输到远程存储前如何执行准备工作 [28]

▼ 如何规划要审计的对象及内容

开始之前 如果要实现非全局区域，请在使用此过程之前查看“规划区域中的审计” [24]。

1. **确定审计策略。**

   缺省情况下，仅启用 cnt 策略。

   使用 auditconfig -lspolicy 命令查看可用策略选项的说明。

- 有关策略选项的影响，请参见“了解审计策略” [29]。
- 有关 cnt 策略的影响，请参见“同步事件和异步事件的审计策略” [107]。
- 要设置审计策略，请参见如何更改审计策略 [44]。

2. **确定是否要修改事件到类的映射。**
   几乎所有情况下，缺省映射便已够用。但是，如果添加新类、更改类定义或确定特定系统调用的记录没有用处，可能需要修改事件到类的映射。
   有关示例，请参见如何更改审计事件的类成员身份 [49]。

3. **确定要预选的审计类。**
   添加审计类或更改缺省类的最佳时间是在用户登录到系统之前。
   在 auditconfig 命令中使用 -setflags 和 -setnaflags 选项预选的审计类适用于所有用户和进程。可以针对成功、失败或两者预选类。
   有关审计类的列表，请阅读 /etc/security/audit_class 文件。

4. **确定系统范围预选的用户修改。**
   如果您确定应当以不同的方式从系统对某些用户进行审计，则可以为个别用户或权限配置文件修改 audit_flags 安全属性。如果已显式为用户设置了审计标志，或者为用户指定了包含显式审计标志的权限配置文件，则会修改用户预选掩码。
   有关过程，请参见如何配置用户审计特征 [40]。要了解哪些是有效的审计标志值，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“所指定权限的搜索顺序”。

5. **决定如何管理 audit_warn 电子邮件别名。**
   每当审计系统检测到需要管理干预的情况时，就会运行 audit_warn 脚本。缺省情况下，audit_warn 脚本会向 audit_warn 别名发送电子邮件，并向控制台发送消息。
   要设置别名，请参见如何配置 audit_warn 电子邮件别名 [47]。

6. **确定收集审计记录的格式和位置。**
   您有三个选择。

   - 缺省情况下，在本地存储二进制审计记录。缺省存储目录为 /var/audit。要进一步配置 audit_binfile 插件，请参见如何为审计文件创建 ZFS 文件系统 [66]。
   - 使用 audit_remote 插件将二进制审计记录以流方式传输到远程的受保护系统信息库。您必须指定记录的接收者。有关要求，请参见“管理远程系统信息库” [17]。有关过程，请参见如何向远程系统信息库发送审计文件 [72]。
   - 使用 audit_syslog 插件将审计记录摘要发送到 syslog。有关过程，请参见如何配置 syslog 审计日志 [78]。

     有关二进制和 syslog 格式的比较，请参见“审计日志” [15]。

7. **确定向管理员发出有关收缩磁盘空间的警告的时间。**

注 - 此步骤仅适用于 audit_binfile 插件。

当审计文件系统上的磁盘空间低于最低空闲空间百分比或软限制时，审计服务将会切换到下一个可用审计目录。然后，服务将发送一条警告，指出已超过软限制。

要了解如何设置最低空闲空间百分比，请参见例 4-7 “设置警告的软限制”。

8. **决定当所有审计目录已满时需要执行的操作。**

注 - 此步骤仅适用于 audit_binfile 插件。

在缺省配置中，audit_binfile 插件处于活动状态，并且设置了 cnt 策略。在此配置中，内核审计队列已满时，系统将继续工作。系统会对丢弃的审计记录进行计数，但是不会记录事件。要获得更大的安全性，可以禁用 cnt 策略，然后启用 ahlt 策略。当异步事件无法放入审计队列时，ahlt 策略会停止系统。

但是，如果 audit_binfile 队列已满，而另一活动插件的队列未满，则内核队列将继续向未满的插件发送记录。audit_binfile 队列可再次接收记录时，审计服务将恢复向其发送记录。

有关 cnt 和 ahlt 策略选项的讨论，请参见“同步事件和异步事件的审计策略” [107]。要了解如何配置这些策略选项，请参见例 3-10 “设置 ahlt 审计策略选项”。

注 - 如果至少一个插件的队列正在接收审计记录，则不会触发 cnt 或 ahlt 策略。

规划审计记录的磁盘空间

audit_binfile 插件可创建审计迹。审计迹需要专用的文件空间。此空间必须可用且安全。对于初始存储，系统使用 /var/audit 文件系统。您可以为审计文件配置其他审计文件系统。以下过程介绍了规划审计迹存储时必须解决的问题。

▼ 如何规划审计记录的磁盘空间

开始之前 如果要实现非全局区域，请在使用此过程之前完成“规划区域中的审计” [24]。

该过程假定您使用的是 audit_binfile 插件。

1. **确定站点所需的审计量。**

针对审计迹平衡磁盘空间可用性和站点的安全需求。

有关如何在保持站点安全的同时降低空间需求，以及如何设计审计存储的指南，请参见“控制审计成本” [31]和“有效审计” [32]。

有关实际步骤，请参见“审计记录的卷过大” [96]、如何压缩专用文件系统上的审计文件 [57]和例 5-4 “合并和减少审计文件”。

2. **确定要审计的系统并配置它们的审计文件系统。**
   创建计划使用的所有文件系统的列表。有关配置准则，请参见“存储和管理审计迹” [17]和 `auditreduce(1M)` 手册页。要指定审计文件系统，请参见如何为审计迹指定审计空间 [69]。

3. **同步所有系统上的时钟。**
   有关更多信息，请参见“确保时间戳可靠” [17]。

准备以流方式将审计记录传输到远程存储

`audit_remote` 插件将二进制的审计迹发送到 ARS，采用的格式与 `audit_binfile` 插件向本地审计文件进行写入时采用的格式相同。`audit_remote` 插件使用 `libgss` 库对 ARS 进行验证，并且使用一种 GSS-API 机制保护传输的私密性和完整性。有关参考信息，请参见《在 Oracle Solaris 11.2 中管理 Kerberos 和其他验证服务 》中的“什么是 Kerberos 服务？”以及《在 Oracle Solaris 11.2 中管理 Kerberos 和其他验证服务 》中的“Kerberos 实用程序”。

当前唯一受支持的 GSS-API 机制是 `kerberosv5`。有关更多信息，请参见 `mech(4)` 手册页。

▼ 以流方式将审计记录传输到远程存储前如何执行准备工作

注 - 如果您有一个 Kerberos 领域，并且在该领域内配置了一个可识别的审计远程服务器 (Audit Remote Server, ARS) 和所有被审计系统，则可以跳过此过程。如何配置审计文件的远程系统信息库 [74]和如何向远程系统信息库发送审计文件 [72]中涵盖了用来配置 ARS 和被审计系统的步骤。

要验证是否配置了 Kerberos 领域，请发送以下命令。该示例输出指示系统未安装 Kerberos。

```
# pkg info system/security/kerberos-5
pkg: info: no packages matching these patterns are installed on the system.
```

开始之前 该过程假定您使用的是 `audit_remote` 插件。

1. **安装主 KDC（Key Distribution Center，密钥分发中心）软件包：**
   您可以使用将用作 ARS 的系统，或者使用邻近的系统。ARS 会向主 KDC 发送大量的验证通信流量。

```
# pkg install pkg:/system/security/kerberos-5
```

在主 KDC 上，可以使用 Kerberos `kdcmgr` 和 `kadmin` 命令来管理领域。有关更多信息，请参见 `kdcmgr`(1M) 和 `kadmin`(1M) 手册页。

2. **在将向 ARS 发送审计记录的每一个被审计系统上，安装主 KDC 软件包。**

```
# pkg install pkg:/system/security/kerberos-5
```

该软件包包含 `kclient` 命令。在这些系统上，运行 `kclient` 命令来与 KDC 进行连接。有关更多信息，请参见 `kclient`(1M) 手册页。

3. **同步 KDC 领域中的时钟。**
如果被审计系统与 ARS 之间的时钟相位差太大，则尝试连接时会失败。在建立连接后，ARS 上的本地时间决定了所存储的审计文件的名称，如“二进制审计文件名称约定” [109]中所述。
有关时钟的更多信息，请参见“确保时间戳可靠” [17]。

了解审计策略

审计策略确定本地系统审计记录的特征。可以使用 `auditconfig` 命令设置这些策略。有关更多信息，请参见 `auditconfig`(1M) 手册页。

缺省情况下，会禁用大多数审计策略选项以最大程度地减少存储需求和系统处理需求。这些选项是审计服务的属性，并确定在系统引导时生效的策略。有关更多信息，请参见 `auditconfig`(1M) 手册页。

参考下表确定启用一个或多个审计策略选项而造成的额外开销是否与站点的需求相适应。

表 2-2 审计策略选项的影响

策略名称	说明	策略注意事项
`ahlt`	该策略仅适用于异步事件。禁用后，该策略允许在不生成审计记录的情况下完成事件。 启用后，该策略会在审计队列已满时停止系统。需要管理干预才能清除审计队列、为审计记录提供空间，以及重新引导系统。只能在全局区域中启用该策略。该策略影响所有区域。	当系统可用性比安全性更重要时，适合禁用该选项。 在安全性极为重要的环境中，适合启用该选项。有关更全面的介绍，请参见“同步事件和异步事件的审计策略” [107]。
`arge`	禁用后，该策略将省略 `execve` 审计记录中已执行程序的环境变量。 启用后，该策略会将已执行程序的环境变量添加到 `execve` 审计记录。与禁用该策略的情况	与启用该选项的情况相比，禁用该选项后收集的信息要少很多。有关比较，请参见如何审计用户执行的所有命令 [51]。

策略名称	说明	策略注意事项
	相比，生成的审计记录包含的详细信息要多得多。	审计少量用户时，适合启用该选项。不确定 ex 审计类的程序中使用的环境变量时，也可以使用此选项。
argv	禁用后，该策略将省略 execve 审计记录中已执行程序的参数。 启用后，该策略会将已执行程序的参数添加到 execve 审计记录。与禁用该策略的情况相比，生成的审计记录包含的详细信息要多得多。	与启用该选项的情况相比，禁用该选项后收集的信息要少很多。有关比较，请参见如何审计用户执行的所有命令 [51]。 审计少量用户时，适合启用该选项。如果您有理由确信 ex 审计类中的异常程序正在运行，也可以使用此选项。
cnt	禁用后，该策略将阻止用户或应用程序运行。由于审计队列已满而导致审计记录无法添加到审计迹时，会发生阻止。 启用后，该策略允许在不生成审计记录的情况下完成事件。该策略将保留丢弃的审计记录数。	在安全性极为重要的环境中，适合禁用该选项。 当系统可用性比安全性更重要时，适合启用该选项。有关更全面的介绍，请参见“同步事件和异步事件的审计策略” [107]。
group	禁用后，该策略不会在审计记录中添加组列表。 启用后，该策略会在每条审计记录中添加组列表作为特殊标记。	禁用该选项通常可以满足站点的安全要求。 当您需要审计主题所属的补充组时，启用该选项很有用。
path	禁用后，该策略会在每条审计记录中最多记录一条在系统调用期间使用的路径。 启用后，该策略会将与审计事件结合使用的每条路径记录到每条审计记录中。	禁用该选项最多在审计记录中放置一条路径。 启用该选项会将系统调用期间使用的每个文件名或路径输入到审计记录中，作为 path 标记。
perzone	禁用后，该策略针对每个系统只维护一个审计配置。全局区域中运行一个审计服务。如果预选 zonename 审计标记，可在审计记录中找到特定区域中的审计事件。 启用后，该策略会为每个区域维护单独的审计配置、审计队列和审计日志。审计服务运行在各个区域中。只能在全局区域中启用该策略。	当您没有特殊理由为每个区域维护单独的审计日志、队列和守护进程时，禁用该选项很有用。 当仅通过检查带有 zonename 审计标记的审计记录无法有效监视系统时，启用该选项很有用。
public	禁用后，如果预选了文件读取，则该策略不会将公共对象的只读事件添加到审计迹。包含只读事件的审计类包括 fr、fa 和 cl。 启用后，如果预选了适当的审计类，则该策略会记录公共对象的每个只读审计事件。	禁用该选项通常可以满足站点的安全要求。 启用该选项的作用很小。
seq	禁用后，该策略不会将序列号添加到每条审计记录中。 启用后，该策略会将序列号添加到每条审计记录中。sequence 标记保留序列号。	当审计顺利进行时，禁用该选项便足以满足要求。 启用 cnt 策略后，适合启用该选项。使用 seq 策略可以确定废弃数据的时间。或者，可以使用 auditstat 命令查看丢弃的记录。
trail	禁用后，该策略不会将 trailer 标记添加到审计记录中。 启用后，该策略会将 trailer 标记添加到每条审计记录中。	禁用该选项会创建一条较小的审计记录。 启用该选项会使用 trailer 标记清楚地标记每条审计记录的末尾。trailer 标记经常与

策略名称	说明	策略注意事项
		sequence 标记结合使用。使用 trailer 标记有助于恢复损坏的审计迹。
zonename	禁用后，该策略不会在审计记录中包括 zonename 标记。	无需跟踪各个区域中的审计行为时，禁用该选项很有用。
	启用后，该策略会在每条审计记录中包括 zonename 标记。	要通过根据区域后选记录来分离和比较各个区域中的审计行为时，启用该选项很有用。

控制审计成本

由于审计会占用系统资源，因此必须控制记录的详细程度。决定要审计的内容时，请考虑以下审计成本：

- 延长处理时间产生的成本
- 分析审计数据产生的成本

如果使用缺省插件 audit_binfile，您还必须考虑审计数据的存储成本。

延长审计数据处理时间产生的成本

延长处理时间产生的成本是审计成本中最不重要的部分。在执行计算密集型任务（如图像处理、复杂计算等）时一般不会进行审计。另外，如果使用 audit_binfile 插件，审计管理员可以将后选任务从被审计系统移动到专用于分析审计数据的系统。最后，除非预选了内核事件，否则，审计服务对系统性能没有明显的影响。

分析审计数据产生的成本

分析成本大致上与收集的审计数据量成正比。分析成本包括合并与查看审计记录所需的时间，

对于 audit_binfile 插件所收集的记录，成本还包括归档记录及其支持名称服务数据库以及妥善保存记录所需的时间。支持数据库包括 groups、hosts 和 passwd。

生成的记录越少，分析审计迹所需的时间就越少。“存储审计数据产生的成本” [32]和“有效审计” [32]部分介绍了如何高效地执行审计。有效的审计可以减少审计数据量，同时提供足够的覆盖范围以实现站点的安全目标。

存储审计数据产生的成本

如果使用 audit_binfile 插件，存储成本是最主要的审计成本。审计数据量取决于以下各项：

- 用户数
- 系统数
- 使用量
- 所需的可追溯与可定责程度

由于上述因素随站点不同而不同，因此没有公式可以预先确定为审计数据存储预留的磁盘空间量。可以使用下列信息作为参考：

- 了解审计类

  配置审计之前，应该了解类中包含的事件类型。可以更改审计事件到类的映射来优化审计记录的收集。
- 精心预选审计类，以减少生成的记录量。

  完全审计（即，使用 all 类）会很快填满磁盘空间。即使是简单的任务（例如编译某个程序）也可能会生成很大的审计文件。一个大小中等的程序在一分钟之内就可能生成数以千计的审计记录。

  例如，省略 file_read 审计类 fr 可以显著减少审计量。通过选择仅针对失败操作进行审计，有时也可以减少审计量。例如，与针对所有 file_read 事件进行审计相比，针对失败的 file_read 操作进行审计（即，使用 -fr）而生成的记录会少很多。
- 如果使用 audit_binfile 插件，有效的审计文件管理也很重要。例如，您可以压缩一个专用于审计文件的 ZFS 文件系统。
- 确立站点审计的思路。

  根据站点所需的可追溯程度以及管理的用户类型等度量，建立自己的理念。

有效审计

以下方法可帮助您在更有效地进行审计的同时实现组织的安全目标。

- 对于尽可能多的审计类，仅针对用户和角色而不是系统范围预选这些类。
- 一次只能对特定百分比的用户进行随机审计。
- 如果 audit_binfile 插件处于活动状态，可通过过滤、合并以及压缩文件来减少审计文件的磁盘存储要求。制订对文件进行归档、将文件传送到可移动介质和脱机存储文件的过程。
- 实时监视审计数据有无异常行为。
  - audit_syslog 插件 – 您可以扩展已开发的管理和分析工具，以处理 syslog 文件中的审计记录。

- `audit_binfile` 插件 – 您可以设置针对某些活动监视审计迹的过程。可以编写一个脚本，以便检测到异常事件时，触发自动提升对特定用户或特定系统的审计。

  例如，可以编写执行以下操作的脚本：

  1. 在被审计系统上监视审计文件的创建。
  2. 使用 `tail` 命令处理审计文件。

     通过 `praudit` 命令对 `tail -0f` 命令进行管道输出，可以在生成记录时产生审计记录流。有关更多信息，请参见 tail(1) 手册页。
  3. 分析此流以查看是否存在异常消息类型或其他指示符，并将分析结果提供给审计程序。

     或者，可以使用脚本来触发自动响应。
  4. 经常监视审计文件系统，以查看是否有新的 `not_terminated` 审计文件出现。
  5. 当等待中的 `tail` 进程的文件不再被写入信息时，终止这些进程。

♦ ♦ ♦ 第 3 章

3 管理审计服务

本章提供用于帮助您在 Oracle Solaris 系统上配置和管理审计的过程。本章涵盖以下任务：

- “审计服务的缺省配置” [35]
- “配置审计服务” [38]
- “定制要审计的内容” [51]
- “在区域中配置审计服务” [59]
- “示例：配置 Oracle Solaris 审计” [62]

此外，以下各章介绍了其他审计管理任务：

- 第 4 章 监视系统活动
- 第 5 章 使用审计数据
- 第 6 章 分析和解决审计服务问题

有关审计服务的概述，请参见第 1 章 关于 Oracle Solaris 中的审计。有关规划建议，请参见第 2 章 规划审计。有关参考信息，请参见第 7 章 审计参考。

审计服务的缺省配置

审计服务具有一个缺省配置，安装 Oracle Solaris 11.2 之后，即可在全局区域中运行该审计服务。无需额外的操作即可启用该服务或将其配置为可用服务。具有缺省配置的审计服务记录以下操作：

- 登录和注销操作
- 使用 `su` 命令
- 屏幕锁定和解除屏幕锁定操作

由于该服务的缺省配置不会影响系统性能，因此不需要为了性能要求而禁用该服务。

假如您拥有与审计相关的对应权限（如 "Audit Review"（审计查看）权限配置文件中的权限），您可以查看审计日志。这些日志存储在 `/var/audit/`*hostname* 中。您可以使

用 praudit 和 auditreduce 命令查看这些文件。有关更多信息，请参见“显示审计迹数据” [81]。

本章中后面的各节提供有关在缺省配置无法满足需要时定制审计服务配置的说明。

显示审计服务缺省值

审计服务由以下参数控制：

- 可归属事件和无归属事件的类
- 审计策略
- 审计插件
- 队列控制

要显示审计服务缺省值，通常可使用 auditconfig -get* 子命令。该子命令显示以星号 (*) 表示的参数的当前配置，如 -getflags -getpolicy 或 -getqctrl。要显示有关无归属事件的类的信息，请使用 auditconfig -getnaflags 子命令。

有关 auditconfig 命令的更多信息，请参见 auditconfig(1M) 手册页。

注 - 要显示审计服务配置，您必须是指定有 "Audit Configuration"（审计配置）或 "Audit Control"（审计控制）权限配置文件的管理员。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

以下示例显示了用于显示缺省审计配置设置的相应命令语法。

例 3-1 显示缺省事件类

在本示例中，两个子命令分别用于显示可归属事件和无归属事件的预选类。要查看哪些事件指定给了某一类以及正在记录哪些事件，请运行 auditrecord -c *class* 命令。

```
# auditconfig -getflags
active user default audit flags = lo(0x1000,0x1000)
configured user default audit flags = lo(0x1000,0x1000)
```

lo 是 login/logout 审计类的标志。掩码输出的格式为 (*success*,*failure*)。

```
# auditconfig -getnaflags
active non-attributable audit flags = lo(0x1000,0x1000)
configured non-attributable audit flags = lo(0x1000,0x1000)
```

例 3-2 显示缺省审计策略

```
$ auditconfig -getpolicy
configured audit policies = cnt
```

```
active audit policies = cnt
```

活动策略是当前策略，但审计服务并不存储该策略值。已配置策略由审计服务存储，因此重新启动审计服务时会恢复该策略。

例 3-3 显示缺省审计插件

```
$ auditconfig -getplugin
Plugin: audit_binfile
Attributes: p_dir=/var/audit;p_fsize=0;p_minfree=1;

Plugin: audit_syslog (inactive)
Attributes: p_flags=;

Plugin: audit_remote (inactive)
Attributes: p_hosts=;p_retries=3;p_timeout=5;
```

缺省情况下 `audit_binfile` 插件处于活动状态。

例 3-4 显示审计队列控制

```
$ auditconfig -getqctrl
no configured audit queue hiwater mark
no configured audit queue lowater mark
no configured audit queue buffer size
no configured audit queue delay
active audit queue hiwater mark (records) = 100
active audit queue lowater mark (records) = 10
active audit queue buffer size (bytes) = 8192
active audit queue delay (ticks) = 20
```

活动队列控制是内核当前使用的队列控制。字符串 `no configured` 表示系统使用的是缺省值。

启用和禁用审计服务

缺省情况下将启用审计服务。如果设置了 `perzone` 审计策略，域管理员必须根据需要在各个非全局区域中启用、刷新或禁用审计服务。如果未设置 `perzone` 审计策略，从全局区域中启用、刷新或禁用审计服务时，也会对所有非全局区域进行相应的操作。

要禁用或启用审计服务，您必须是指定有 "Audit Control"（审计控制）权限配置文件的管理员。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

要禁用审计服务，请使用以下命令：

```
# audit -t
```

要启用审计服务，请使用以下命令：

```
# audit -s
```

要验证审计服务是否正在运行，请使用以下命令：

```
# auditconfig -getcond
audit condition = auditing
```

如果设置了 perzone 审计策略，则必须在启用了审计的非全局区域中执行此验证。

有关更多信息，请参见 audit(1M) 和 auditd(1M) 手册页。

配置审计服务

在网络上启用审计之前，可以修改缺省值以满足站点审计要求。最佳做法是在第一批用户登录之前尽可能定制审计配置。

如果已实现区域，则可以选择从全局区域审计所有区域或单独审计非全局区域。有关概述的信息，请参见“审计和 Oracle Solaris 区域” [104]。有关规划的信息，请参见“规划区域中的审计” [24]。有关过程的信息，请参见“在区域中配置审计服务” [59]。

要配置审计服务，通常使用 auditconfig 子命令。使用这些子命令设置的配置适用于整个系统。

- auditconfig -get* 显示以星号 (*) 表示的参数的当前配置，如“显示审计服务缺省值” [36]中的示例所示。
- auditconfig -set* 指定以星号 (*) 表示的参数的值，如 -setflags、-setpolicy 或 -setqctrl。要配置无归属事件的类，请使用 auditconfig setnaflags 子命令。

您也可以定制适用于用户或配置文件而不是整个系统的审计。为每个用户预选的审计类是由 audit_flags 安全属性指定的。这些用户特定值以及系统的预选类确定用户的审计掩码，如“进程审计特征” [108]中所述。

通过基于每个用户而非基于每个系统预选类，有时可以降低审计对系统性能的影响。此外，您可能还需要审计与系统稍有不同的特定用户。

要配置适用于用户或配置文件的审计，可使用以下命令：

- usrattr 显示为用户设置的 audit_flags 值。缺省情况下，仅针对系统范围的设置审计用户。
- usermod -K 设置适用于用户的标志。
- profile 设置适用于配置文件的标志。

有关 userattr 命令的说明，请参见 userattr(1) 手册页。有关 audit_flags 关键字的说明，请参见 user_attr(4) 手册页。

以下任务列表列出了配置审计的过程。所有任务都是可选的。

表 3-1 配置审计服务的任务列表

任务	说明	参考
选择要审计的事件。	预选系统范围的审计类。如果是可归属事件，则针对此事件审计所有用户。	如何预选审计类 [39]
选择要针对特定用户审计的事件。	设置系统范围的审计类中特定于用户的差异。	如何配置用户审计特征 [40]
指定审计策略。	定义站点所需的其他审计数据。	如何更改审计策略 [44]
指定队列控制。	修改缺省缓冲区大小、队列中的审计记录以及两次将审计记录写入缓冲区之间的间隔。	如何更改审计队列控制 [46]
创建 `audit_warn` 电子邮件别名。	定义需要关注审计服务时接收电子邮件警告的人员。	如何配置 `audit_warn` 电子邮件别名 [47]
配置审计日志。	为每个插件配置审计记录的位置。	“配置审计日志” [65]
添加审计类。	通过创建用来保存关键事件的新审计类来减少审计记录数目。	如何添加审计类 [48]
更改事件到类的映射。	通过更改事件到类的映射来减少审计记录数目。	如何更改审计事件的类成员身份 [49]

▼ 如何预选审计类

预选包含要监视事件的审计类。不记录预选类之外的事件。

开始之前 您必须是指定有 "Audit Configuration"（审计配置）权限配置文件的管理员。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

1. **确定当前预选类。**

```
# auditconfig -getflags
...

# auditconfig -getnaflags
,,,
```

有关输出的说明，请参见“显示审计服务缺省值” [36]。

2. **预选可归属类。**

```
# auditconfig -setflags lo,ps,fw
user default audit flags = ps,lo,fw(0x101002,0x101002)
```

此命令审计 `login/logout`、`process start/stop` 和 `file write` 类中的事件是成功还是失败。

注 - auditconfig -setflags 命令会替换当前预选类，因此必须指定要预选的所有类。

3. 预选无归属类。

   na 类包含其他事件之间的 PROM、引导和无归属挂载。

```
# auditconfig -setnaflags lo,na
non-attributable audit flags = lo,na(0x1400,0x1400)
```

   lo 和 na 是 -setnaflags 选项唯一有用的两个参数。

注 - auditconfig -setnaflags 命令会替换当前预选类，因此必须指定要预选的所有类。

▼ 如何配置用户审计特征

可将这些特定于用户且借助于本过程设置的审计特征与系统的预选类结合使用。它们能共同决定用户的审计掩码，如“进程审计特征” [108]中所示。

开始之前 您必须承担 root 角色。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

1. （可选） 显示当前针对现有用户所选的审计类。

   a. 显示用户的列表。

```
# who
adoe    pts/1        Oct 10 10:20 (:0.0)
adoe    pts/2        Oct 10 10:20 (:0.0)
jdoe    pts/5        Oct 12 12:20 (:0.0)
jdoe    pts/6        Oct 12 12:20 (:0.0)
...
```

   b. 显示各个用户的 **audit_flags** 属性值。

```
# userattr audit_flags adoe
# userattr audit_flags jdoe
```

2. 在 **user_attr** 或 **prof_attr** 数据库中设置审计标志。

   例如，您可以创建一个权限配置文件，用以定义您的一部分用户的权限。对于指定有该权限配置文件的用户，将以相同的方式对其进行审计。

   - 要为用户设置审计标志，请使用 **usermod** 命令。

```
# usermod -K audit_flags=fw:no jdoe
```

audit_flags 关键字的格式为 *always-audit*:*never-audit*。

always-audit 列出针对此用户审计的审计类。对系统范围类的修改带有插入记号 (^) 前缀。添加到系统范围类的类不带插入记号前缀。

never-audit 列出从不针对用户审计的审计类，即使在系统范围内审计这些审计事件也是如此。对系统范围类的修改带有插入记号 (^) 前缀。

要指定多个审计类，请使用逗号分隔类。有关更多信息，请参见 audit_flags(5) 手册页。

- **要为权限配置文件设置审计标志，请使用 profiles 命令。**

```
# profiles -p "System Administrator"
profiles:System Administrator> set name="Audited System Administrator"
profiles:Audited System Administrator> set always_audit=fw,as
profiles:Audited System Administrator> end
profiles:Audited System Administrator> exit
```

将 "Audited System Administrator"（审计的系统管理员）权限配置文件指定给用户或角色时，按《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“所指定权限的搜索顺序”中所述的搜索顺序针对这些标志审计该用户或角色。

例 3-5 更改对单个用户进行审计的事件

在本示例中，所有用户的审计预选掩码如下：

```
# auditconfig -getflags
active user default audit flags = ss,lo(0x11000,0x11000)
configured user default audit flags = ss,lo(0x11000,0x11000)
```

除管理员以外没有任何用户登录。

为了降低 AUE_PFEXEC 审计事件对系统资源的影响，管理员将不在系统级别审计此事件。相反，管理员将为用户 jdoe 预选 pf 类。pf 类是在例 3-15 “创建新的审计类” 中创建的。

```
# usermod -K audit_flags=pf:no jdoe
```

userattr 命令显示添加内容。

```
# userattr audit_flags jdoe
pf:no
```

用户 jdoe 登录时，jdoe 的审计预选掩码为 audit_flags 值和系统缺省值的组合。289 是 jdoe 的登录 shell 的 PID。

```
# auditconfig -getpinfo 289
```

```
audit id = jdoe(1234)
process preselection mask = ss,pf,lo(0x0100000008011000,0x0100000008011000)
terminal id (maj,min,host) = 242,511,example1(192.168.160.171)
audit session id = 103203403
```

例 3-6 修改单个用户的审计预选例外

在本示例中，所有用户的审计预选掩码如下：

```
# auditconfig -getflags
active user default audit flags = ss,lo(0x11000,0x11000)
configured user default audit flags = ss,lo(0x11000,0x11000)
```

除管理员以外没有任何用户登录。

管理员决定不为 jdoe 用户收集失败的 ss 事件。

```
# usermod -K audit_flags=^-ss:no jdoe
```

userattr 命令显示例外。

```
# userattr audit_flags jdoe
^-ss:no
```

用户 jdoe 登录时，jdoe 的审计预选掩码为 audit_flags 值和系统缺省值的组合。289 是 jdoe 的登录 shell 的 PID。

```
# auditconfig -getpinfo 289
audit id = jdoe(1234)
process preselection mask = +ss,lo(0x11000,0x1000)
terminal id (maj,min,host) = 242,511,example1(192.168.160.171)
audit session id = 103203403
```

例 3-7 审计选定用户，非系统范围的审计

在本示例中，审计四个选定用户在系统上的登录和角色活动。没有为系统预选审计类。

首先，管理员删除所有系统范围的标志。

```
# auditconfig -setflags no
user default audit flags = no(0x0,0x0)
```

然后，管理员为这四个用户预选两个审计类。pf 类是在例 3-15 “创建新的审计类” 中创建的。

```
# usermod -K audit_flags=lo,pf:no jdoe
# usermod -K audit_flags=lo,pf:no kdoe
# usermod -K audit_flags=lo,pf:no pdoe
# usermod -K audit_flags=lo,pf:no zdoe
```

然后，管理员为 root 角色预选 pf 类。

```
# userattr audit_flags root
# rolemod -K audit_flags=lo,pf:no root
# userattr audit_flags root
lo,pf:no
```

为了继续记录未经授权的入侵，管理员将不更改无归属登录的审计。

```
# auditconfig -getnaflags
active non-attributable audit flags = lo(0x1000,0x1000)
configured non-attributable audit flags = lo(0x1000,0x1000)
```

例 3-8 删除用户的审计标志

在以下示例中，管理员删除所有用户特定的审计标志。继续审计当前已登录用户的现有进程。

管理员运行 usermod 命令，但不为 audit_flags 关键字设置任何值。

```
# usermod -K audit_flags= jdoe
# usermod -K audit_flags= kdoe
# usermod -K audit_flags= ldoe
```

然后，管理员验证删除。

```
# userattr audit_flags jdoe
# userattr audit_flags kdoe
# userattr audit_flags ldoe
```

例 3-9 为用户组创建权限配置文件

管理员需要站点的所有管理权限配置文件才能显式审计 pf 类。对于要指定的每个权限配置文件，管理员都会在包含审计标志的 LDAP 中创建特定于站点的版本。

首先，管理员克隆现有权限配置文件，然后更改名称并添加审计标志。

```
# profiles -p "Network Wifi Management" -S ldap
profiles: Network Wifi Management> set name="Wifi Management"
profiles: Wifi Management> set desc="Audited wifi management"
profiles: Wifi Management> set audit_always=pf
profiles: Wifi Management> exit
```

针对要使用的每个权限配置文件重复此过程之后，管理员将列出 Wifi Management（Wifi 管理）配置文件中的信息。

```
# profiles -p "Wifi Management" -S ldap info
name=Wifi Management
desc=Audited wifi management
auths=solaris.network.wifi.config
help=RtNetWifiMngmnt.html
```

```
always_audit=pf
```

▼ 如何更改审计策略

您可能会更改缺省审计策略以记录审计命令的相关详细信息、为每个记录添加区域名称，或满足其他站点安全要求。

开始之前 您必须是指定有 "Audit Configuration"（审计配置）权限配置文件的管理员。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

1. **查看当前审计策略。**

```
$ auditconfig -getpolicy
...
```

有关输出的说明，请参见“显示审计服务缺省值” [36]。

2. **查看可用的策略选项。**

```
$ auditconfig -lspolicy
policy string    description:
ahlt             halt machine if it can not record an async event
all              all policies for the zone
arge             include exec environment args in audit recs
argv             include exec command line args in audit recs
cnt              when no more space, drop recs and keep a cnt
group            include supplementary groups in audit recs
none             no policies
path             allow multiple paths per event
perzone          use a separate queue and auditd per zone
public           audit public files
seq              include a sequence number in audit recs
trail            include trailer token in audit recs
windata_down     include downgraded window information in audit recs
windata_up       include upgraded window information in audit recs
zonename         include zonename token in audit recs
```

注 - 只能在全局区域中设置 `perzone` 和 `ahlt` 策略选项。有关使用特定策略选项的权衡，请参见“了解审计策略” [29]。

3. **启用或禁用选定的审计策略选项。**

```
# auditconfig [ -t ] -setpolicy [prefix]policy[,policy...]
```

-t 可选。创建临时或活动策略。可以设置用于调试或测试的临时策略。

刷新审计服务或通过 `auditconfig -setpolicy` 命令修改策略之前，临时策略起作用。

prefix — *prefix* 值 + 将策略列表添加到当前策略中。*prefix* 值 - 从当前策略中删除策略列表。没有前缀时，将重置审计策略。通过此选项，您可以保留当前审计策略。

policy — 选择要启用或禁用的策略。

例 3-10 设置 `ahlt` 审计策略选项

在此示例中，严格的站点安全性需要使用 `ahlt` 策略。

```
# auditconfig -setpolicy -cnt
# auditconfig -setpolicy +ahlt
```

`ahlt` 策略前的加号 (+) 将策略添加到当前策略设置。没有加号时，`ahlt` 策略将替换所有当前审计策略。

例 3-11 设置临时审计策略

在此示例中，配置了 `ahlt` 审计策略。为了进行调试，管理员将 `trail` 审计策略临时 (-t) 添加到活动策略中 (+trail)。`trail` 策略用于帮助恢复损坏的审计迹。

```
$ auditconfig -setpolicy ahlt
$ auditconfig -getpolicy
configured audit policies = ahlt
active audit policies = ahlt
$ auditconfig -t -setpolicy +trail
configured audit policies = ahlt
active audit policies = ahlt,trail
```

调试完成后，管理员将禁用 `trail` 策略。

```
$ auditconfig -setpolicy -trail
$ auditconfig -getpolicy
configured audit policies = ahlt
active audit policies = ahlt
```

通过运行 `audit -s` 命令刷新审计服务也可以删除此临时策略，以及审计服务中的任何其他临时值。有关其他临时值的示例，请参见如何更改审计队列控制 [46]。

例 3-12 设置 `perzone` 审计策略

在本示例中，将 `perzone` 审计策略添加到全局区域中的现有策略。由于 `perzone` 策略设置是作为永久属性存储的，因此 `perzone` 策略在会话过程中重新启动审计服务后才生效。对于这些区域，策略将在下一次区域引导时可用。

```
$ auditconfig -getpolicy
configured audit policies = cnt
active audit policies = cnt
$ auditconfig -setpolicy +perzone
$ auditconfig -getpolicy
configured audit policies = perzone,cnt
active audit policies = perzone,cnt
```

例 3-13 收集外部审计者的审计记录

在本示例中，管理员为了满足外部审计者的要求而收集审计记录。管理员决定使用审计远程服务器 (Audit Remote Server, ARS) 收集有关管理活动的信息。管理员还收集不归属于用户的操作（如引导）。

管理员设置 ARS。除审计 `cusa` 类外，管理员还向审计配置中添加策略。

```
# auditconfig -setflags cusa
user default audit flags = ex,xa,ua,as,ss,ap,lo,ft(0x80475080,0x80475080)
# auditconfig -setpolicy ahlt,argv,argeauditconfig # auditconfig -getpolicy
configured audit policies = ahlt,arge,argv
active audit policies = ahlt,arge,argv
# auditconfig -setnaflags lo,na
non-attributable audit flags = lo,na(0x1400,0x1400)
```

管理员启用 `audit_remote` 插件并刷新审计服务后，系统即开始收集记录。

▼ 如何更改审计队列控制

审计服务提供审计队列参数的缺省值。可以使用 `auditconfig` 命令检查、永久更改和临时更改这些值。

开始之前 您必须是指定有 "Audit Configuration"（审计配置）权限配置文件的管理员。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

1. **查看审计队列控制的当前值。**

   ```
   $ auditconfig -getqctrl
   ...
   ```

   有关输出的说明，请参见“显示审计服务缺省值” [36]。

2. **修改选定的审计队列控制。**

   有关审计队列控制的示例和说明，请参见 `auditconfig(1M)` 手册页。

   - 要修改部分或全部审计队列控制，请使用 `-setqctrl` 选项。

```
# auditconfig [ -t ] -setqctrl hiwater lowater bufsz interval
```

高水位 (hiwater) 和低水位 (lowater) 的值分别指示暂停和恢复进程的临界点。这些临界点以未传送的审计记录数进行度量。缓冲区大小 (bufsz) 指的是队列的缓冲区大小。Interval 指示生成审计输出的延迟（以时钟周期进行度量）。

例如，将 *interval* 值设置为 10，但不设置其他控制。

```
# auditconfig -setqctrl 0 0 0 10
```

- 要修改特定审计队列控制，请指定其选项。-setqdelay 选项等同于 -setqctrl 0 0 0 *interval*，如 **auditconfig -setqdelay 10** 中所示。

```
# auditconfig [ -t ] -setqhiwater value
# auditconfig [ -t ] -setqlowater value
# auditconfig [ -t ] -setqbufsz value
# auditconfig [ -t ] -setqdelay value
```

例 3-14 将审计队列控制重置为缺省值

管理员设置所有审计队列控制，然后将系统信息库中的 *lowater* 值更改回缺省值。

```
# auditconfig -setqctrl 200 5 10216 10
# auditconfig -setqctrl 200 0 10216 10
configured audit queue hiwater mark (records) = 200
no configured audit queue lowater mark
configured audit queue buffer size (bytes) = 10216
configured audit queue delay (ticks) = 10
active audit queue hiwater mark (records) = 200
active audit queue lowater mark (records) = 5
active audit queue buffer size (bytes) = 10216
active audit queue delay (ticks) = 10
```

随后，管理员将当前会话的 *lowater* 值设置为缺省值。

```
# auditconfig -setqlowater 10
# auditconfig -getqlowater
configured audit queue lowater mark (records) = 10
active audit queue lowater mark (records) = 10
```

▼ 如何配置 audit_warn 电子邮件别名

/etc/security/audit_warn 脚本将生成邮件，通知管理员可能需要关注的审计事件。您可以定制脚本，可以将邮件发送给 root 以外的帐户。

如果设置了 perzone 策略，非全局区域的管理员必须在非全局区域中配置 audit_warn 电子邮件别名。

开始之前 您必须是指定有 solaris.admin.edit/etc/security/audit_warn 授权的管理员。缺省情况下，只有 root 角色具有此授权。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

- **配置 audit_warn 电子邮件别名。**

  选择以下选项之一：

  - 在 audit_warn 脚本中使用其他电子邮件帐户替换 audit_warn 电子邮件别名。

    将脚本的 ADDRESS 行中的 audit_warn 电子邮件别名更改为其他地址：

    ```
    #ADDRESS=audit_warn            # standard alias for audit alerts
    ADDRESS=audadmin               # role alias for audit alerts
    ```

    注 - 有关修改审计配置文件的效果的信息，请参见“审计配置文件和软件包” [104]。

  - 将 audit_warn 电子邮件重定向到其他电子邮件帐户。

    将 audit_warn 电子邮件别名添加到相应的邮件别名文件中。您可以将别名添加到本地 /etc/mail/aliases 文件或者添加到名称空间中的 mail_aliases 数据库。如果将 root 和 audadmin 电子邮件帐户添加为 audit_warn 电子邮件别名的成员，/etc/mail/aliases 项将类似于以下示例：

    ```
    audit_warn: root,audadmin
    ```

    然后，运行 newaliases 命令为 aliases 文件重新生成随机访问数据库。

    ```
    # newaliases
    /etc/mail/aliases: 14 aliases, longest 10 bytes, 156 bytes total
    ```

▼ 如何添加审计类

创建您自己的审计类时，可以只将需要针对您所在站点审计的审计事件存放到该类中。这一策略可以减少所收集的记录数，并减少审计迹中的无用数据。

在一个系统上添加类时，将此更改复制到正在审计的所有系统中。最佳做法是在第一个用户登录之前创建审计类。

有关修改审计配置文件的效果的信息，请参见“审计配置文件和软件包” [104]。

提示 - 在 Oracle Solaris 中，您可以创建自己的包含文件的软件包，并用您的站点定制文件替换 Oracle Solaris 软件包。当您将软件包中的 preserve 属性设置为 true 时，pkg 子命令（例如 verify、fix、revert，等等）将相对于您的软件包运行。有关更多信息，请参见 pkg(1) 和 pkg(5) 手册页。

开始之前 为您的唯一条目选择空闲位。验证哪些位可供客户在 `/etc/security/audit_class` 文件中使用。

您必须是指定有 `solaris.admin.edit/etc/security/audit_class` 授权的管理员。缺省情况下，只有 `root` 角色具有此授权。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

1. （可选） 保存 **`audit_class`** 文件的备份副本。

```
# cp /etc/security/audit_class /etc/security/audit_class.orig
```

2. 在 **`audit_class`** 文件中添加新项。
每一项都具有以下格式：

```
0x64bitnumber:flag:description
```

有关字段的说明，请参见 `audit_class(4)` 手册页。有关现有类的列表，请阅读 `/etc/security/audit_class` 文件。

例 3-15 创建新的审计类

本示例创建一个类来保存角色中执行的管理命令。`audit_class` 文件中添加的项如下所示：

```
0x0100000000000000:pf:profile command
```

该条目创建新的 `pf` 审计类。例 3-16 “将现有审计事件映射到新类” 显示如何填充新的审计类。

故障排除 如果您定制了 `audit_class` 文件，请确保直接指定给用户或权限配置文件的任何审计标志与新的审计类一致。`audit_flags` 值不是 `audit_class` 文件的子集时发生错误。

▼ 如何更改审计事件的类成员身份

可能需要更改审计事件的类成员身份来减小现有审计类的大小，或者将事件放置在它自己的类中。

注意 - 切勿注释掉 `audit_event` 中的事件。该文件供 `praudit` 命令用来读取二进制审计文件。归档审计文件可能包含该文件中列出的事件。

在一个系统上重新配置审计事件到类的映射时，将此更改复制到正在审计的所有系统中。最佳做法是第一个用户登录之前更改事件到类映射。

注 - 有关修改审计配置文件的效果的信息，请参见“审计配置文件和软件包” [104]。

提示 - 在 Oracle Solaris 中，您可以创建自己的包含文件的软件包，并用您的站点定制文件替换 Oracle Solaris 软件包。当您将软件包中的 `preserve` 属性设置为 `true` 时，`pkg` 子命令（例如 verify、fix、revert，等等）将相对于您的软件包运行。有关更多信息，请参见 `pkg`(1) 和 `pkg`(5) 手册页。

开始之前 您必须是指定有 `solaris.admin.edit/etc/security/audit_event` 授权的管理员。缺省情况下，只有 `root` 角色具有此授权。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

1. （可选） 保存 **`audit_event`** 文件的备份副本。

```
# cp /etc/security/audit_event /etc/security/audit_event.orig
```

2. 通过更改事件的 *class-list* 来更改特定事件所属的类。

   每一项都具有以下格式：

   number:*name*:*description*:*class-list*

number	审计事件 ID。
name	审计事件的名称。
description	通常是触发创建审计记录的系统调用或可执行文件。
class-list	以逗号分隔的审计类列表。

例 3-16 将现有审计事件映射到新类

本示例中将现有审计事件映射到在例 3-15 “创建新的审计类” 中创建的新类。缺省情况下，`AUE_PFEXEC` 审计事件映射到多个审计类。通过创建新类，管理员可以审计 `AUE_PFEXEC` 事件而无需审计其他类中的事件。

```
# grep pf /etc/security/audit_class
0x0100000000000000:pf:profile command
# grep AUE_PFEXEC /etc/security/audit_event
116:AUE_PFEXEC:execve(2) with pfexec enabled:ps,ex,ua,as,cusa
# pfedit /etc/security/audit_event
#116:AUE_PFEXEC:execve(2) with pfexec enabled:ps,ex,ua,as,cusa
116:AUE_PFEXEC:execve(2) with pfexec enabled:pf
# auditconfig -setflags lo,pf
user default audit flags = pf,lo(0x0100000000001000,0x0100000000001000)
```

定制要审计的内容

以下任务列表列出了根据您的特定需要配置审计的过程。

表 3-2 定制审计任务列表

任务	说明	参考
对用户在系统上执行的所有操作进行审计。	针对每个命令审计一个或多个用户。	如何审计用户执行的所有命令 [51]
更改正在记录的审计事件，并使更改影响现有会话。	更新用户的预选掩码。	如何更新已登录用户的预选掩码 [55]
找到对特定文件的修改。	审计文件修改，然后使用 auditreduce 命令找出特定文件。	如何找到对特定文件更改的审计记录 [53]
使审计文件使用更少的文件系统空间。	使用 ZFS 配额和压缩。	如何压缩专用文件系统上的审计文件 [57]
从 audit_event 文件中删除审计事件。	正确更新 audit_event 文件。	如何阻止审计特定事件 [56]

▼ 如何审计用户执行的所有命令

作为站点安全策略的一部分，有些站点会需要 root 帐户和管理角色运行的所有命令的审计记录。一些站点可能会需要所有用户运行的所有命令的审计记录。此外，站点可能还会要求记录命令参数和环境。

开始之前 要预选审计类和设置审计策略，您必须是指定有 "Audit Configuration"（审计配置）权限配置文件的管理员。要将审计标志指定给用户、角色和权限配置文件，您必须承担 root 角色。

1. **显示 lo 和 ex 类的用户级别事件信息。**

   ex 类审计对 exec（） 和 execve（） 函数的所有调用。

   lo 类审计登录、注销和屏幕锁定。以下输出列出了 ex 和 lo 类中的所有事件。

```
% auditconfig -lsevent | grep " lo "
AUE_login                       6152 lo login - local
AUE_logout                      6153 lo logout
AUE_telnet                      6154 lo login - telnet
AUE_rlogin                      6155 lo login - rlogin
AUE_rshd                        6158 lo rsh access
AUE_su                          6159 lo su
AUE_rexecd                      6162 lo rexecd
AUE_passwd                      6163 lo passwd
AUE_rexd                        6164 lo rexd
AUE_ftpd                        6165 lo ftp access
AUE_ftpd_logout                 6171 lo ftp logout
```

```
AUE_ssh                         6172 lo login - ssh
AUE_role_login                  6173 lo role login
AUE_newgrp_login                6212 lo newgrp login
AUE_admin_authenticate          6213 lo admin login
AUE_screenlock                  6221 lo screenlock - lock
AUE_screenunlock                6222 lo screenlock - unlock
AUE_zlogin                      6227 lo login - zlogin
AUE_su_logout                   6228 lo su logout
AUE_role_logout                 6229 lo role logout
AUE_smbd_session                6244 lo smbd(1m) session setup
AUE_smbd_logoff                 6245 lo smbd(1m) session logoff
AUE_ClientConnect               9101 lo client connection to x server
AUE_ClientDisconnect            9102 lo client disconn. from x server

% auditconfig -lsevent | egrep " ex |,ex |ex,"
AUE_EXECVE                        23 ex,ps execve(2)
```

2. 审计 **lo** 和 **ex** 类。

   - 要审计管理角色的这些类，请修改角色的安全属性。
     在以下示例中，root 是一个角色。站点已创建 sysadm、auditadm 和 netadm 三个角色。针对 ex 和 lo 类中的成功事件和失败事件审计所有角色。

     ```
     # rolemod -K audit_flags=lo,ex:no root

     # rolemod -K audit_flags=lo,ex:no sysadm

     # rolemod -K audit_flags=lo,ex:no auditadm

     # rolemod -K audit_flags=lo,ex:no netadm
     ```

   - 要审计所有用户的这些类，请设置系统范围的标志。

     ```
     # auditconfig -setflags lo,ex
     ```

     输出内容类似如下：

     ```
     header,129,2,AUE_EXECVE,,mach1,2010-10-14 12:17:12.616 -07:00
     path,/usr/bin/ls
     attribute,100555,root,bin,21,320271,18446744073709551615
     subject,jdoe,root,root,root,root,2486,50036632,82 0 mach1
     return,success,0
     ```

3. 指定要记录的、与命令用法有关的其他信息。

   - 要记录命令参数，请添加 **argv** 策略。

     ```
     # auditconfig -setpolicy +argv
     ```

     exec_args 标记用于记录命令参数：

     ```
     header,151,2,AUE_EXECVE,,mach1,2010-10-14 12:26:17.373 -07:00
     ```

```
path,/usr/bin/ls
attribute,100555,root,bin,21,320271,18446744073709551615
exec_args
,2,ls,/etc/security
subject,jdoe,root,root,root,root,2494,50036632,82 0 mach1
return,success,0
```

- 要记录命令的运行环境，请添加 **arge** 策略。

  ```
  # auditconfig -setpolicy +arge
  ```

  exec_env 标记用于记录命令环境：

  ```
  header,1460,2,AUE_EXECVE,,mach1,2010-10-14 12:29:39.679 -07:00
  path,/usr/bin/ls
  attribute,100555,root,bin,21,320271,18446744073709551615
  exec_args,2,ls,/etc/security
  exec_env
  ,49,MANPATH=/usr/share/man,USER=jdoe,GDM_KEYBOARD_LAYOUT=us,EDITOR=gedit,
  LANG=en_US.UTF-8,GDM_LANG=en_US.UTF-8,PS1=#,GDMSESSION=gnome,SESSIONTYPE=1,SHLVL=2,
  HOME=/home/jdoe,LOGNAME=jdoe,G_FILENAME_ENCODING=@locale,UTF-8,
  PRINTER=example-dbl,...,_=/usr/bin/ls
  subject,jdoe,root,root,root,root,2502,50036632,82 0 mach1
  return,success,0
  ```

▼ 如何找到对特定文件更改的审计记录

如果您的目标是记录对有限数目的文件（如 /etc/passwd 和 /etc/default 目录下的文件）的写入情况，请使用 auditreduce 命令找到这些文件。

开始之前 root 角色可以执行此过程中的所有任务。

如果您的组织中的管理权限是分散的，请注意以下事项：

- 具有 "Audit Configuration"（审计配置）权限配置文件的管理员可以运行 auditconfig 命令。
- 具有 "Audit Review"（审计查看）权限配置文件的管理员可以运行 auditreduce 命令。
- 只有 root 角色可以指定审计标志。

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

1. 执行以下步骤之一以审计文件更改。

   - 审计 fw 类。

     将 fw 类添加到用户或角色的审计标志时生成的记录少于将该类添加到系统范围的审计预选掩码时生成的记录。执行下列步骤之一：

- 将 `fw` 类添加到特定角色。

```
# rolemod -K audit_flags=fw:no root
# rolemod -K audit_flags=fw:no sysadm
# rolemod -K audit_flags=fw:no auditadm
# rolemod -K audit_flags=fw:no netadm
```

- 将 `fw` 类添加到系统范围的标志。

```
# auditconfig -getflags
active user default audit flags = lo(0x1000,0x1000)
configured user default audit flags = lo(0x1000,0x1000)

# auditconfig -setflags lo,fw
user default audit flags = lo,fw(0x1002,0x1002)
```

- 审计成功的文件写入。

  审计成功事件时生成的记录少于同时审计失败事件和成功事件时生成的记录。执行下列步骤之一：

  - 将 `+fw` 标志添加到特定角色。

```
# rolemod -K audit_flags=+fw:no root
# rolemod -K audit_flags=+fw:no sysadm
# rolemod -K audit_flags=+fw:no auditadm
# rolemod -K audit_flags=+fw:no netadm
```

  - 将 `+fw` 标志添加到系统范围的标志。

```
# auditconfig -getflags
active user default audit flags = lo(0x1000,0x1000)
configured user default audit flags = lo(0x1000,0x1000)

# auditconfig -setflags lo,+fw
user default audit flags = lo,+fw(0x1002,0x1000)
```

2. 使用 **auditreduce** 命令获取特定文件的审计记录。

```
# auditreduce -o file=/etc/passwd,/etc/default -O filechg
```

   `auditreduce` 命令在审计迹中搜索所有 `file` 参数实例。该命令创建一个后缀为 `filechg` 的二进制文件，此文件包含了包括所需文件路径的所有记录。有关 `-o file=` *pathname* 选项的语法，请参见 `auditreduce(1M)` 手册页。

3. 使用 **praudit** 命令读取 **filechg** 文件。

```
# praudit *filechg
```

▼ 如何更新已登录用户的预选掩码

本过程介绍如何审计已登录用户的系统范围审计预选掩码的更改。您通常可以通过指示用户注销并重新登录来完成此任务。或者，以指定有 "Process Management"（进程管理）权限配置文件的角色，可以使用 `kill` 命令手动终止活动会话。新的会话将继承新的预选掩码。

但是，终止用户会话可能是不可行的。作为备选方案，您可以使用 `auditconfig` 命令动态更改各个已登录用户的预选掩码。

在以下过程中，假定已通过运行以下命令将系统范围审计预选掩码从 `lo` 更改为 `lo,ex`：

```
# auditconfig -setflags lo,ex
```

开始之前 您必须是指定有 "Audit Configuration"（审计配置）权限配置文件的管理员。要终止用户会话，您必须是指定有 "Process Management"（进程管理）权限配置文件的管理员。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

1. 列出登录的一般用户及其进程 ID。

```
# who -a
jdoe  - vt/2         Jan 25 07:56  4:10   1597   (:0)
jdoe  + pts/1        Jan 25 10:10   .     1706   (:0.0)
...
jdoe  + pts/2        Jan 25 11:36  3:41   1706   (:0.0)
```

2. 为便于以后进行比较，显示每个用户的预选掩码。

```
# auditconfig -getpinfo 1706
audit id = jdoe(1234)
process preselection mask = lo(0x1000,0x1000)
terminal id (maj,min,host) = 9426,65559,mach1(192.168.123.234)
audit session id = 103203403
```

3. 通过运行下面的一个或多个命令，修改相应的预选掩码：

```
# auditconfig -setpmask 1706 lo,ex            /* for this process */

# auditconfig -setumask jdoe lo,ex            /* for this user */

# auditconfig -setsmask 103203403 lo,ex       /* for this session */
```

4. 验证用户的预选掩码是否已更改。

   例如，检查更改掩码之前存在的进程。

```
# auditconfig -getpinfo 1706
audit id = jdoe(1234)
```

```
process preselection mask = ex,lo(0x40001000,0x40001000)
terminal id (maj,min,host) = 9426,65559,mach1(192.168.123.234)
audit session id = 103203403
```

▼ 如何阻止审计特定事件

出于维护目的，有时某个站点需要阻止对事件进行审计。

开始之前 您必须承担 root 角色。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

1. **将相关事件的类更改为 no 类。**

   注 - 有关修改审计配置文件的效果的信息，请参见“审计配置文件和软件包” [104]。

   例如，事件 26 和 27 属于 pm 类。

```
## audit_event file
...
25:AUE_VFORK:vfork(2):ps
26:AUE_SETGROUPS:setgroups(2):pm
27:AUE_SETPGRP:setpgrp(2):pm
28:AUE_SWAPON:swapon(2):no
...
```

   将这些事件更改为 no 类。

```
## audit_event file
...
25:AUE_VFORK:vfork(2):ps
26:AUE_SETGROUPS:setgroups(2):no
27:AUE_SETPGRP:setpgrp(2):no
28:AUE_SWAPON:swapon(2):no
...
```

   如果当前正在审计 pm 类，现有会话仍将审计事件 26 和 27。要停止审计这些事件，必须遵循如何更新已登录用户的预选掩码 [55]中的说明更新用户的预选掩码。

   注意 - 切勿注释掉 audit_event 中的事件。该文件供 praudit 命令用来读取二进制审计文件。归档审计文件可能包含该文件中列出的事件。

2. **刷新内核事件。**

```
# auditconfig -conf
Configured 283 kernel events.
```

▼ 如何压缩专用文件系统上的审计文件

审计文件可以增大。您可以设置文件大小的上限，如例 4-3 “限制 audit_binfile 插件的文件大小”中所示。在本过程中，使用压缩来减小大小。

开始之前 您必须是指定有 "ZFS File System Management"（ZFS 文件系统管理）和 "ZFS Storage Management"（ZFS 存储管理）权限配置文件的管理员。通过后一个配置文件可以创建存储池。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

1. **将 ZFS 文件系统专用于审计文件。**

   有关过程，请参见如何为审计文件创建 ZFS 文件系统 [66]。

2. **使用以下选项之一压缩 ZFS 存储池。**

   使用这两个选项均可压缩审计文件系统。刷新审计服务后，将显示压缩率。

   在以下示例中，ZFS 池 `auditp/auditf` 为数据集。

   - **使用缺省压缩算法。**

     ```
     # zfs set compression=on auditp/auditf
     # audit -s
     # zfs get compressratio auditp/auditf
     NAME           PROPERTY       VALUE  SOURCE
     auditp/auditf  compressratio  4.54x  -
     ```

   - **使用更高压缩算法。**

     ```
     # zfs set compression=gzip-9 auditp/auditf
     # zfs get compression auditp/auditf
     NAME           PROPERTY     VALUE     SOURCE
     auditp/auditf  compression  gzip-9    local
     ```

     `gzip-9` 压缩算法导致文件占用的空间比缺省压缩算法 `lzjb` 少三分之一。有关更多信息，请参见《在 Oracle Solaris 11.2 中管理 ZFS 文件系统 》中的第 5 章 “管理 Oracle Solaris ZFS 文件系统”。

3. **刷新审计服务。**

   ```
   # audit -s
   ```

4. **（可选） 验证新的压缩设置。**

   例如，如果使用高压缩算法，信息可能与以下内容类似：

   ```
   # zfs get compressration auditp/auditf
   NAME           PROPERTY       VALUE  SOURCE
   auditp/auditf  compressratio  16.89x  -
   ```

▼ 如何审计 FTP 和 SFTP 文件传输

FTP 服务创建其文件传输的日志。通过预选 `ft` 审计类可以审计 `ssh` 协议下运行的 SFTP 服务。可以审计到这两种服务的登录。

注 - 有关如何记录 FTP 服务的命令和文件传输的信息，请参见 `proftpd`(8) 手册页。

有关可用的日志记录选项，请阅读 ProFTPD Logging (http://www.proftpd.org/docs/howto/Logging.html)（ProFTPD 日志记录）。

- **根据您要审计的是 SFTP 还是 FTP，输入以下任一命令。**
  - 要记录 `sftp` 访问和文件传输，请编辑 `ft` 类。

    `ft` 类包含以下 SFTP 事务：

    ```
    % auditrecord -c ft
    file transfer: chmod ...
    file transfer: chown ...
    file transfer: get ...
    file transfer: mkdir ...
    file transfer: put ...
    file transfer: remove ...
    file transfer: rename ...
    file transfer: rmdir ...
    file transfer: session start ...
    file transfer: session end ...
    file transfer: symlink ...
    file transfer: utimes
    ```

  - 要记录对 FTP 服务器的访问，请审计 `lo` 类。

    如以下示例输出所示，登录和注销 `proftpd` 守护进程都会生成审计记录。

    ```
    % auditrecord -c lo | more
    ...
    FTP server login
    program     proftpd             See in.ftpd(1M)
    event ID    6165                AUE_ftpd
    class       lo                  (0x0000000000001000)
    header
    subject
    [text]                      error message
    return

    FTP server logout
    program     proftpd             See in.ftpd(1M)
    ```

```
event ID    6171                AUE_ftpd_logout
class       lo                  (0x0000000000001000)
header
subject
return
...
```

在区域中配置审计服务

审计服务审计整个系统，包括区域中的审计事件。安装了非全局区域的系统可以以相同方式审计所有区域，也可基于每个区域配置审计。有关更多信息，请参见“规划区域中的审计” [24]。

当对非全局区域完全像对全局区域一样进行审计时，非全局区域的管理员可能无法访问审计记录。而且，全局区域管理员可以修改非全局区域中用户的审计预选掩码。

当分别审计各个非全局区域时，审计记录的可见范围是相应的非全局区域以及从该非全局区域根产生的全局区域。

▼ 如何配置以相同方式审计所有区域

此过程可以实现以相同方式审计所有区域。该方法需要的计算机开销和管理资源最少。

开始之前 您必须承担 `root` 角色。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

1. **配置全局区域的审计。**

   完成“配置审计服务” [38]中的任务，但要注意以下例外：

   - 不要启用 `perzone` 审计策略。
   - 设置 `zonename` 策略。此策略将区域名称添加到每个审计记录。

     ```
     # auditconfig -setpolicy +zonename
     ```

2. **如果修改了审计配置文件，请将这些文件从全局区域复制到每个非全局区域。**

   如果修改了 `audit_class` 或 `audit_event` 文件，请采用以下两种方式之一复制它：

   - 可以回送挂载文件。
   - 可以复制文件。

   非全局区域必须在运行中。

- 将更改的 **audit_class** 和 **audit_event** 文件作为回送文件系统 (**lofs**) 进行挂载。

  a. 从全局区域中，中止非全局区域。

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

  b. 为在全局区域中修改的每个审计配置文件创建一个只读回送挂载。

  ```
  # zonecfg -z non-global-zone
  zone: add fs
  zone/fs: set special=/etc/security/audit-file
  zone/fs: set dir=/etc/security/audit-file
  zone/fs: set type=lofs
  zone/fs: add options [ro,nodevices,nosetuid]
  zone/fs: commit
  zone/fs: end
  zone: exit
  #
  ```

  c. 要使更改生效，请引导该非全局区域。

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

  之后，如果在全局区域中修改了审计配置文件，请重新引导每个区域以刷新非全局区域中回送挂载的文件。

- 复制文件。

  a. 在全局区域中，列出每个非全局区域中的 **/etc/security** 目录。

  ```
  # ls /zone/zonename/root/etc/security/
  ```

  b. 将更改的 **audit_class** 和 **audit_event** 文件复制到每个区域的 **/etc/security** 目录。

  ```
  # cp /etc/security/audit-file /zone/zonename/root/etc/security/audit-file
  ```

  之后，如果在全局区域中更改其中一个文件，则必须将更改的文件复制到非全局区域。

当在全局区域中重新启动审计服务时，或者重新引导非全局区域时，将对非全局区域进行审计。

例 3-17 在区域中挂载作为回送挂载的审计配置文件

在本示例中，系统管理员修改了 `audit_class`、`audit_event` 和 `audit_warn` 文件。

`audit_warn` 文件仅在全局区域中读取，因此不必将其挂载到非全局区域。

在系统 machine1 中，管理员创建了两个非全局区域：machine1–webserver 和 machine1–appserver。管理员已经完成了对审计配置文件的修改。如果管理员以后修改文件，必须重新引导区域以重新读取回送挂载。

```
# zoneadm -z machine1-webserver halt
# zoneadm -z machine1-appserver halt
# zonecfg -z machine1-webserver
webserver: add fs
webserver/fs: set special=/etc/security/audit_class
webserver/fs: set dir=/etc/security/audit_class
webserver/fs: set type=lofs
webserver/fs: add options [ro,nodevices,nosetuid]
webserver/fs: commit
webserver/fs: end
webserver: add fs
webserver/fs: set special=/etc/security/audit_event
webserver/fs: set dir=/etc/security/audit_event
webserver/fs: set type=lofs
webserver/fs: add options [ro,nodevices,nosetuid]
webserver/fs: commit
webserver/fs: end
webserver: exit
#

# zonecfg -z machine1-appserver
appserver: add fs
appserver/fs: set special=/etc/security/audit_class
appserver/fs: set dir=/etc/security/audit_class
appserver/fs: set type=lofs
appserver/fs: add options [ro,nodevices,nosetuid]
appserver/fs: commit
appserver/fs: end
appserver: exit
```

重新引导非全局区域时，audit_class 和 audit_event 文件在区域中为只读。

▼ 如何配置每区域审计

此过程使每个区域的管理员可以在他们的各自区域中控制审计服务。有关策略选项的完整列表，请参见 auditconfig(1M) 手册页。

开始之前 要配置审计，您必须是指定有 "Audit Configuration"（审计配置）权限配置文件的管理员。要启用审计服务，您必须是指定有 "Audit Control"（审计控制）权限配置文件的管理员。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

1. **在全局区域中，配置审计。**

a. 完成“配置审计服务” [38]中的任务。

b. 添加 **perzone** 审计策略。有关命令，请参见例 3-12 “设置 **perzone** 审计策略”。

注 - 您无需在全局区域中启用审计服务。

2. 在计划审计的每个非全局区域中，配置审计文件。

a. 完成“配置审计服务” [38]中的任务。

b. 不要配置系统范围的审计设置。

具体来说，不要将 `perzone` 或 `ahlt` 策略添加到非全局区域中。

3. 在您的区域中启用审计。

```
myzone# audit -s
```

例 3-18 在非全局区域中禁用审计

此示例适用于已设置了 `perzone` 审计策略的情况。`noaudit` 区域的管理员禁用该区域的审计。

```
noauditzone # auditconfig -getcond
audit condition = auditing
noauditzone # audit -t
noauditzone # auditconfig -getcond
audit condition = noaudit
```

示例：配置 Oracle Solaris 审计

本节提供了如何配置和实现 Oracle Solaris 审计的示例。它首先根据特定需求和要求配置服务的不同属性。完成配置后，启动审计服务以使配置设置生效。在每次需要修正现有审计配置以满足新的要求时，请遵循此示例中的相同操作顺序：

1. 配置审计参数。
2. 刷新审计服务。
3. 验证新的审计配置。

- 首先，管理员添加临时策略。

```
# auditconfig -t -setpolicy +zonename
# auditconfig -getpolicy
```

```
configured audit policies = ahlt,arge,argv,perzone
active audit policies = ahlt,arge,argv,perzone,zonename
```

- 然后，管理员指定队列控制。

```
# auditconfig -setqctrl 200 20 0 0
# auditconfig -getqctrl
configured audit queue hiwater mark (records) = 200
configured audit queue lowater mark (records) = 20
configured audit queue buffer size (bytes) = 8192
configured audit queue delay (ticks) = 20
active audit queue hiwater mark (records) = 200
active audit queue lowater mark (records) = 20
active audit queue buffer size (bytes) = 8192
active audit queue delay (ticks) = 20
```

- 随后，管理员指定插件属性。
  - 对于 audit_binfile 插件，管理员将删除 qsize 值。

```
# auditconfig -getplugin audit_binfile
Plugin: audit_binfile
Attributes: p_dir=/audit/sys1.1,/var/audit;
p_minfree=2;p_fsize=4G;
Queue size: 200
# auditconfig -setplugin audit_binfile  "" 0
# auditconfig -getplugin audit_binfile
Plugin: audit_binfile
Attributes: p_dir=/audit/sys1.1,/var/audit
p_minfree=2;p_fsize=4G;
```

  - 对于 audit_syslog 插件，管理员将指定要发送到 syslog 的成功登录和注销事件以及失败的可执行文件。此插件的 qsize 设置为 150。

```
# auditconfig -setplugin audit_syslog active p_flags=+lo,-ex 150
# auditconfig -getplugin audit_syslog
auditconfig -getplugin audit_syslog
Plugin: audit_syslog
Attributes: p_flags=+lo,-ex;
Queue size: 150
```

  - 管理员不配置也不使用 audit_remote 插件。
- 然后，管理员刷新审计服务并验证配置。
  - 不再设置临时 zonename 策略。

```
# audit -s
# auditconfig -getpolicy
configured audit policies = ahlt,arge,argv,perzone
active audit policies = ahlt,arge,argv,perzone
```

  - 队列控制保留原样。

```
# auditconfig -getqctrl
configured audit queue hiwater mark (records) = 200
configured audit queue lowater mark (records) = 20
configured audit queue buffer size (bytes) = 8192
configured audit queue delay (ticks) = 20
active audit queue hiwater mark (records) = 200
active audit queue lowater mark (records) = 20
active audit queue buffer size (bytes) = 8192
active audit queue delay (ticks) = 20
```

- audit_binfile 插件没有指定的队列大小。audit_syslog 插件具有指定的队列大小。

```
# auditconfig -getplugin
Plugin: audit_binfile
Attributes: p_dir=/var/audit;p_fsize=4G;p_minfree=2;

Plugin: audit_syslog
Attributes: p_flags=+lo,-ex;
Queue size: 50
...
```

♦ ♦ ♦ 第 4 章

4 监视系统活动

本章提供的过程可帮助您配置审计日志，通过这些日志可以监视系统中的活动。此外，以下各章介绍了其他审计管理任务：

- 第 3 章 管理审计服务
- 第 5 章 使用审计数据
- 第 6 章 分析和解决审计服务问题

有关审计服务的概述，请参见第 1 章 关于 Oracle Solaris 中的审计。有关规划建议，请参见第 2 章 规划审计。有关参考信息，请参见第 7 章 审计参考。

配置审计日志

audit_binfile 和 audit_syslog 这两个审计插件可以创建本地审计日志。以下任务说明如何配置这些日志。

配置审计日志

以下任务列表列出了为各种插件配置审计日志的过程。配置 audit_binfile 插件的日志是可选的。管理员必须配置其他插件的日志。

表 4-1 配置审计日志任务列表

任务	说明	参考
为 audit_binfile 插件添加本地存储	为审计文件创建附加磁盘空间，并使用文件权限来保护它们	如何为审计文件创建 ZFS 文件系统 [66]
为 audit_binfile 插件指定存储	标识用于二进制审计记录的目录	如何为审计迹指定审计空间 [69]
配置审计记录向远程系统的流传输	使您能够通过受保护的机制将审计记录发送到远程系统信息库	如何向远程系统信息库发送审计文件 [72]

任务	说明	参考
配置审计文件的远程存储	使您可以在远程系统上接收审计记录	如何配置审计文件的远程系统信息库 [74]
为 audit_syslog 插件配置存储。	使您能够以文本格式将审计事件流式传输到 syslog。	如何配置 syslog 审计日志 [78]

▼ 如何为审计文件创建 ZFS 文件系统

以下过程说明如何为审计文件创建 ZFS 池以及相应的文件系统和挂载点。缺省情况下，/var/audit 文件系统为 audit_binfile 插件保存审计文件。

开始之前 您必须是指定有 "ZFS File System Management"（ZFS 文件系统管理）和 "ZFS Storage Management"（ZFS 存储管理）权限配置文件的管理员。通过后一个配置文件可以创建存储池。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

1. **确定所需的磁盘空间量。**

   每个主机至少指定 200 MB 磁盘空间。但是，由于所需审计量决定磁盘空间需求，所需的磁盘空间可能远大于此数字。

   注 - 缺省类预选在 /var/audit 中创建文件，对于 lo 类中记录的每个事件实例（如登录、注销或角色承担），这些文件增长约 80 字节。

   ---

2. **创建镜像 ZFS 存储池。**

   zpool create 命令创建一个存储池作为 ZFS 文件系统的容器。有关更多信息，请参见《在 Oracle Solaris 11.2 中管理 ZFS 文件系统 》中的第 1 章 “Oracle Solaris ZFS 文件系统（介绍）”。

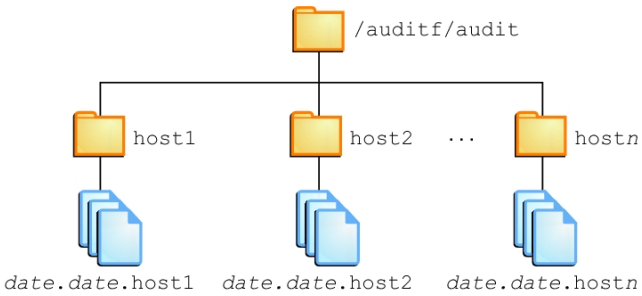
   ```
   # zpool create audit-pool mirror disk1 disk2
   ```

   例如，从两个磁盘（c3t1d0 和 c3t2d0）创建 auditp 池并镜像这两个磁盘。

   ```
   # zpool create auditp mirror c3t1d0 c3t2d0
   ```

3. **为审计文件创建 ZFS 文件系统和挂载点。**

   使用一个命令创建文件系统和挂载点。创建时，文件系统已挂载。例如，下图显示了按主机名存储的审计迹存储。



注 - 如果计划加密文件系统，必须在创建文件系统时对其进行加密。有关示例，请参见例 4-1 “为审计文件创建加密文件系统”。

加密需要进行管理。例如，挂载时需要口令短语。有关更多信息，请参见《在 Oracle Solaris 11.2 中管理 ZFS 文件系统 》中的“加密 ZFS 文件系统”。

```
# zfs create -o mountpoint=/mountpoint audit-pool/mountpoint
```

例如，为 `auditf` 文件系统创建 `/audit` 挂载点。

```
# zfs create -o mountpoint=/audit auditp/auditf
```

4. 为审计文件创建 ZFS 文件系统。

```
# zfs create -p auditp/auditf/system
```

例如，为 `sys1` 系统创建未加密的 ZFS 文件系统。

```
# zfs create -p auditp/auditf/sys1
```

5. （可选） 为审计文件创建其他文件系统。

创建其他文件系统的一个原因是阻止审计溢出。可以设置每个文件系统的 ZFS 配额，如步骤 8 中所示。达到每个配额时，`audit_warn` 电子邮件别名会通知您。要释放空间，可以将关闭的审计文件移动到远程服务器。

```
# zfs create -p auditp/auditf/sys1.1

# zfs create -p auditp/auditf/sys1.2
```

6. 保护父审计文件系统。

为池中的所有文件系统将以下 ZFS 属性设置为 `off`：

```
# zfs set devices=off auditp/auditf

# zfs set exec=off auditp/auditf
```

```
# zfs set setuid=off auditp/auditf
```

7. 压缩池中的审计文件。
   通常，压缩是在 ZFS 中的文件系统级别设置的。但是，由于此池中的所有文件系统都包含审计文件，因此会在顶级数据集为池设置压缩。

   ```
   # zfs set compression=on auditp
   ```

   另请参见《在 Oracle Solaris 11.2 中管理 ZFS 文件系统 》中的“ZFS 压缩、重复数据删除和加密属性之间的交互”。

8. 设置配额。
   可以设置父文件系统和/或后代文件系统上的配额。如果设置父审计文件系统上的配额，后代文件系统上的配额将强加其他限制。

   a. 设置父审计文件系统上的配额。
      在以下示例中，`auditp` 池中的两个磁盘都达到配额时，`audit_warn` 脚本将通知审计管理员。

      ```
      # zfs set quota=510G auditp/auditf
      ```

   b. 设置后代审计文件系统上的配额。
      在以下示例中，达到 `auditp/auditf/`*system* 文件系统的配额时，`audit_warn` 脚本将通知审计管理员。

      ```
      # zfs set quota=170G auditp/auditf/sys1

      # zfs set quota=170G auditp/auditf/sys1.1

      # zfs set quota=165G auditp/auditf/sys1.2
      ```

9. 对于大型池，限制审计文件的大小。
   缺省情况下，审计文件可以增长到池的大小。为了便于管理，限制审计文件的大小。请参见例 4-3 “限制 `audit_binfile` 插件的文件大小”。

例 4-1 为审计文件创建加密文件系统

为了符合站点安全要求，管理员需要执行以下步骤：

1. 创建新的 ZFS 池来存储加密的审计日志（如有必要）。
2. 生成密钥。
3. 在启用加密的情况下，创建用于存储审计日志的审计文件系统并设置挂载点。
4. 将审计配置为使用加密目录。
5. 刷新审计服务以应用新的配置设置。

```
# zpool create auditp mirror disk1 disk2
```

```
# pktool genkey keystore=file outkey=/filename keytype=aes keylen=256

# zfs create -o encryption=aes-256-ccm \
-o keysource=raw,file:///filename \
-o compression=on -o mountpoint=/audit auditp/auditf

# auditconfig -setplugin audit_binfile p_dir=/audit/

# audit -s
```

您必须备份和保护存储密钥的文件，如本例中的 *filename*。

管理员在 `auditf` 文件系统下创建其他文件系统时，也将加密这些后代文件系统。

例 4-2 在 `/var/audit` 目录上设置配额

在本示例中，管理员在缺省审计文件系统上设置配额。达到此配额时，`audit_warn` 脚本将警告审计管理员。

```
# zfs set quota=252G rpool/var/audit
```

▼ 如何为审计迹指定审计空间

在本过程中，使用 `audit_binfile` 插件的属性将其他磁盘空间指定给审计迹。

开始之前 您必须是指定有 "Audit Configuration"（审计配置）权限配置文件的管理员。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

1. **确定 `audit_binfile` 插件的属性。**

   阅读 `audit_binfile (5)` 手册页的 `OBJECT ATTRIBUTES` 部分。

```
# man audit_binfile

...
OBJECT ATTRIBUTES
The p_dir attribute specifies where the audit files will be created.
The directories are listed in the order in which they are to be used.

The p_minfree attribute defines the percentage of free space that the
audit system requires before the audit daemon invokes the audit_warn
script.
```

```
The p_fsize attribute defines the maximum size that an audit
file can become before it is automatically closed and a new
audit file is opened. ... The format of the p_fsize value can
be specified as an exact value in bytes or in a human-readable
form with a suffix of  B,  K, M, G, T, P, E, Z (for bytes,
kilobytes, megabytes, gigabytes, terabytes,  petabytes, exabytes,
or zettabytes, respectively). Suffixes of KB, MB, GB, TB, PB, EB,
and ZB are also accepted.
```

2. **要将目录添加到审计迹，请指定 `p_dir` 属性。**

   缺省文件系统为 `/var/audit`。

   ```
   # auditconfig -setplugin audit_binfile p_dir=/audit/sys1.1,/var/audit
   ```

   上述命令将 `/audit/sys1.1` 文件系统设置为审计文件的主目录，将缺省 `/var/audit` 文件系统设置为辅助目录。在此方案中，`/var/audit` 是最后考虑采用的目录。要成功完成此配置，`/audit/sys1.1` 文件系统必须存在。

   在如何为审计文件创建 ZFS 文件系统 [66]中创建了类似的文件系统。

3. **刷新审计服务。**

   `auditconfig -setplugin` 命令设置已配置的值。该值为审计服务的属性，因此刷新或重新启动该服务时会恢复该值。刷新或重新启动审计服务时，已配置的值变为活动状态。有关配置的活动值的信息，请参见 `auditconfig(1M)` 手册页。

   ```
   # audit -s
   ```

例 4-3 限制 `audit_binfile` 插件的文件大小

在以下示例中，二进制审计文件的大小设置为特定大小。大小是以兆字节为单位指定的。

```
# auditconfig -setplugin audit_binfile p_fsize=4M

# auditconfig -getplugin audit_binfile
Plugin: audit_binfile
Attributes: p_dir=/var/audit;p_fsize=4M;p_minfree=1;
```

缺省情况下，审计文件大小可以无限制地增长。为了创建较小的审计文件，管理员指定文件大小上限为 4 MB。达到大小限制时，审计服务将创建新文件。管理员刷新审计服务后，文件大小限制开始生效。

```
# audit -s
```

例 4-4 指定日志轮转时间

在下面的示例中，为审计文件设置了时间限制。时间限制按小时、天、周、月或年来指定。

```
# auditconfig -setplugin audit_binfile "p_age=1w"

# auditconfig -getplugin audit_binfile
Plugin: audit_binfile
Attributes: p_dir=/var/audit;p_age=1w;
Queue size: 200
```

缺省情况下，审计文件没有时间限制。该文件将无限期保持打开状态，直到外部操作使文件轮转为止。管理员将文件的时间限制设置为一周，超过这个时间限制将打开新的审计文件。要实现新的时间限制，管理员需刷新审计服务。

```
# audit -s
```

例 4-5 指定对审计插件的多个更改

在以下示例中，具有高吞吐量和大型 ZFS 池的系统上的管理员将更改 `audit_binfile` 插件的队列大小、二进制文件大小和软限制警告。管理员允许审计文件增长到 4 GB，在 ZFS 池剩余 2% 时收到警告，并将允许的队列大小翻倍。缺省队列大小是内核审计队列的高水位标志 `100`，如 `active audit queue hiwater mark (records) = 100` 中所示。还将审计文件的时间限制设置为 2 周。

```
# auditconfig -getplugin audit_binfile
Plugin: audit_binfile
Attributes: p_dir=/var/audit;p_fsize=2G;p_minfree=1;

# auditconfig -setplugin audit_binfile \
      "p_minfree=2;p_fsize=4G;p_age=2w" 200

# auditconfig -getplugin audit_binfile
Plugin: audit_binfile
Attributes: p_dir=/var/audit;p_fsize=4G;p_minfree=2;p_age=2w;
Queue size: 200
```

管理员刷新审计服务后，更改的规范开始生效。

```
# audit -s
```

例 4-6 删除审计插件的队列大小

在以下示例中，将删除 `audit_binfile` 插件的队列大小。

```
# auditconfig -getplugin audit_binfile
Plugin: audit_binfile
Attributes: p_dir=/var/audit;p_fsize=4G;p_minfree=2;
Queue size: 200

# auditconfig -setplugin audit_binfile "" 0

# auditconfig -getplugin audit_binfile
Plugin: audit_binfile
Attributes: p_dir=/var/audit;p_fsize=4G;p_minfree=2;
```

空引号 ("") 将保留当前属性值。最后的 0 将插件的队列大小设置为缺省值。

管理员刷新审计服务后，插件的 qsize 规范中的更改开始生效。

```
# audit -s
```

例 4-7 设置警告的软限制

在本示例中，设置了所有审计文件系统的最低空闲空间级别，以便在文件系统的可用空间只有 2% 时发出警告。

```
# auditconfig -setplugin audit_binfile p_minfree=2
```

缺省百分比为 1 (1)。对于大型 ZFS 池，请适当地选择较低的百分比。例如，16 TB 池的 10% 大约是 16 GB，如果设为此值将在仍剩余大量磁盘空间时向审计管理员发出警告。如果值为 2，将在剩余大约 2 GB 磁盘空间时发送 audit_warn 消息。

audit_warn 电子邮件别名用于接收警告。要设置别名，请参见如何配置 audit_warn 电子邮件别名 [47]。

对于大型池，管理员还会将文件大小限制为 3 GB。

```
# auditconfig -setplugin audit_binfile p_fsize=3G
```

管理员刷新审计服务后，插件的 p_minfree 和 p_fsize 规范开始生效。

```
# audit -s
```

▼ 如何向远程系统信息库发送审计文件

在本过程中，使用 audit_remote 插件的属性将审计迹发送到远程审计系统信息库。要在 Oracle Solaris 系统上配置一个远程系统信息库，请参见如何配置审计文件的远程系统信息库 [74]。

开始之前 远程系统信息库上必须具有审计服务的接收者。您必须是指定有 "Audit Configuration"（审计配置）权限配置文件的管理员。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

1. **确定 audit_remote 插件的属性。**

   阅读 audit_remote(5) 手册页的 OBJECT ATTRIBUTES 部分。

   ```
   # man audit_remote

   ...
   OBJECT ATTRIBUTES
   ```

```
The p_hosts attribute specifies the remote servers.
You can also specify the port number and the GSS-API
mechanism.

The p_retries attribute specifies the number of retries for
connecting and sending data. The default is 3.

The p_timeout attribute specifies the number of seconds
in which a connection times out.
```

缺省端口是 `solaris_audit` IANA 指定的端口 `16162/tcp`。缺省机制是 `kerberos_v5`。缺省超时为 5 秒。您还可以指定插件的队列大小。

2. **要指定远程接收系统，请使用 `p_hosts` 属性。**

   在本示例中，接收系统使用一个不同的端口。

   ```
   # auditconfig -setplugin audit_remote \
        p_hosts=ars.example.com:16088:kerberos_v5
   ```

3. **指定要更改的插件的其他属性。**

   例如，以下命令指定所有可选属性的值：

   ```
   # auditconfig -setplugin audit_remote "p_retries=;p_timeout=3" 300
   ```

4. **验证这些值，然后激活插件。**

   例如，以下命令指定并验证插件的值：

   ```
   # auditconfig -getplugin audit_remote
   Plugin: audit_remote (inactive)
   Attributes: p_hosts=ars.example.com:16088:kerberos_v5;p_retries=5;p_timeout=3;
   Queue size: 300

   # auditconfig -setplugin audit_remote active
   ```

5. **刷新审计服务。**

   审计服务在刷新时将读取审计插件更改。

   ```
   # audit -s
   ```

例 4-8 调优审计队列缓冲区大小

在本示例中，`audit_remote` 插件后面的审计队列太满。该被审计系统被配置为对许多类进行审计，并在通过高通信流量的慢速网络传输数据。管理员增大了插件的缓冲区大小，使审计队列可以增长，从而使得在审计记录从队列中移除之前不会超出缓冲区大小限制。

```
audsys1 # auditconfig -setplugin audit_remote "" 1000

audsys1 # audit -s
```

▼ 如何配置审计文件的远程系统信息库

在此过程中，您将配置一个远程系统，即审计远程服务器 (Audit Remote Server, ARS)，以接收并存储来自一个或多个被审计系统的审计记录。然后，您将激活远程服务器上的审计守护进程。

配置分两部分。首先，您将配置底层安全机制以便安全地传输审计数据，也就是说，您将配置 KDC。其次，您将在被审计系统和 ARS 上配置审计服务。此过程阐明了一个包含一个被审计客户机和一个 ARS 的方案，其中，ARS 和 KDC 位于同一服务器上。对于更复杂的方案，可以按类似方式进行配置。前四步骤描述了 KDC 的配置，而最后一步描述了审计服务的配置。

开始之前 请确保您已完成以下操作。您必须承担 `root` 角色。

- 您已承担 `root` 角色。
- 您已安装了 Kerberos 软件包，如以流方式将审计记录传输到远程存储前如何执行准备工作 [28]中所述。
- 与配置了被审计系统的管理员协作，如如何向远程系统信息库发送审计文件 [72]中所述。

1. **如果您的站点尚未配置 KDC，请配置一个 KDC。**

   您需要在一个系统上配置一个 KDC，让被审计系统和 ARS 都能使用，需要有每个系统的主机主体，另外还要有一个 `audit` 服务主体。以下示例说明了 KDC 配置策略：

   ```
   arstore # kdcmgr -a audr/admin -r EXAMPLE.COM create master
   ```

   此命令使用 `audr/admin` 管理主体在 `EXAMPLE.COM` 领域中创建一个主 KDC 并将其启用，然后启动 Kerberos 服务。

2. **验证 KDC 是否可用。**

   有关更多信息，请参见 `kdcmgr(1M)` 手册页。

   ```
   # kdcmgr status

   KDC Status Information
   ----------------------------------------
   svc:/network/security/krb5kdc:default (Kerberos key distribution center)
   State: online since Wed Feb 29 01:59:27 2012
   See: man -M /usr/share/man -s 1M krb5kdc
   See: /var/svc/log/network-security-krb5kdc:default.log
   Impact: None.

   KDC Master Status Information
   ----------------------------------------
   svc:/network/security/kadmin:default (Kerberos administration daemon)
   State: online since Wed Feb 29 01:59:28 2012
   See: man -M /usr/share/man -s 1M kadmind
   See: /var/svc/log/network-security-kadmin:default.log
   ```

```
Impact: None.

Transaction Log Information
-------------------------------------------

Kerberos update log (/var/krb5/principal.ulog)
Update log dump :
Log version # : 1
Log state : Stable
Entry block size : 2048
Number of entries : 13
First serial # : 1
Last serial # : 13
First time stamp : Wed Feb 29 01:59:27 2012
Last time stamp : Mon Mar 5 19:29:28 2012


Kerberos Related File Information
-------------------------------------------
(Displays any missing files)
```

3. 将 **audit** 服务主体添加到 KDC 密钥表文件中。

   您可以通过在 KDC 系统上键入 `kadmin.local` 命令来添加主体。另外，也可以通过使用 `kadmin` 命令并提供口令来远程添加主体。在本示例中，`arstore` 系统在运行 KDC。

   ```
   # kadmin -p audr/admin

   kadmin: addprinc -randkey audit/arstore.example.com@EXAMPLE.COM

   kadmin: ktadd audit/arstore.example.com@EXAMPLE.COM
   ```

4. 在每个被审计系统上添加密钥。

   接收者和发送者都必须有密钥。

   ```
   enigma # kclient

   .. Enter the Kerberos realm:
   EXAMPLE.COM

   .. KDC hostname for the above realm:
   arstore.example.com

   .. Will this client need service keys ? [y/n]:
   y
   ```

5. 在 ARS 上配置审计服务。

   - 要创建一个组以接收来自 Kerberos 领域中任何被审计系统的审计记录，请命名一个连接组。

     ```
     # auditconfig -setremote group create Bank_A
     ```

Bank_A 是一个连接组。因为未定义 hosts 属性，所以此组将接受所有连接，这意味着它是一个通配符组。该 Kerberos 领域中已正确配置了 audit_remote 插件的任何被审计系统都可以访问此 ARS。

- **要限制与此组的连接，请指定可以使用此系统信息库的被审计系统。**

```
# auditconfig -setremote group Bank_A "hosts=enigma.example.com"
```

连接组 Bank_A 现在只接受来自 enigma 系统的连接。来自任何其他主机的连接将被拒绝。

- **为防止此组中的审计文件增长得过大，请设置最大大小。**

```
# auditconfig -setremote group Bank_A "binfile_fsize=4GB"

# auditconfig -getremote
Audit Remote Server
Attributes: listen_address=;login_grace_time=30;max_startups=10;listen_port=0;
Connection group: Bank_A (inactive)
Attributes: binfile_dir=/var/audit;binfile_fsize=4GB;binfile_minfree=1;
hosts=enigma.example.com;
```

6. **在被审计系统上配置审计服务。**

要指定 ARS，请使用 p_hosts 属性。

```
enigma # auditconfig -setplugin audit_remote \
     active p_hosts=arstore.example.com

enigma # auditconfig -getplugin audit_remote
Plugin: audit_remote
Attributes: p_retries=3;p_timeout=5;p_hosts=arstore.example.com;
```

7. **刷新审计服务。**

审计服务在刷新时将读取审计插件更改。

```
# audit -s
```

现在，被审计系统 enigma 与 ARS 之间的连接由 KDC 进行管理。

例 4-9 将审计记录以流方式传输到同一 ARS 上的不同文件位置

此示例扩展了上述过程中的示例。管理员通过创建两个连接组，在 ARS 上按主机对审计记录进行了划分。

来自 audsys1 的审计文件将以流方式传输到此 ARS 上的 Bank_A 连接组。

```
arstore # auditconfig -setremote group create Bank_A

arstore # auditconfig -setremote group active Bank_A "hosts=audsys1" \
```

```
   "hosts=audsys1;binfile_dir=/var/audit/audsys1;binfile_fsize=4M;"
```

来自 audsys2 的审计文件将以流方式传输到 Bank_B 连接组。

```
arstore # auditconfig -setremote group create Bank_B

arstore # auditconfig -setremote group active Bank_B \
"hosts=audsys2;binfile_dir=/var/audit/audsys2;binfile_fsize=4M;"
```

为了更加方便地进行维护，管理员以相同方式设置了其他属性值。

```
arstore # auditconfig -getremote
Audit Remote Server
Attributes: listen_address=;login_grace_time=30;max_startups=10;listen_port=0;

Connection group: Bank_A
Attributes: binfile_dir=/var/audit/audsys1;binfile_fsize=4M;binfile_minfree=1;
hosts=audsys1

Connection group: Bank_B
Attributes: binfile_dir=/var/audit/audsys2;binfile_fsize=4M;binfile_minfree=1;
hosts=audsys2
```

例 4-10 将 ARS 置于与 KDC 不同的系统上

在本示例中，管理员将 ARS 置于与 KDC 不同的一个系统上。首先，管理员创建并配置主 KDC。

```
kserv # kdcmgr -a audr/admin -r EXAMPLE.COM create master

kserv # kadmin.local -p audr/admin

kadmin: addprinc -randkey \
audit/arstore.example.com@EXAMPLE.COM

kadmin: ktadd -t /tmp/krb5.keytab.audit \
     audit/arstore.example.com@EXAMPLE.COM
```

在将 /tmp/krb5.keytab.audit 文件安全地传输到 ARS arstore 后，管理员将该文件移动到了正确的位置。

```
arstore # chown root:root krb5.keytab.audit

arstore # chmod 600 krb5.keytab.audit

arstore # mv krb5.keytab.audit /etc/krb5/krb5.keytab
```

除了重写文件，管理员还可以选择在 ARS 上使用 ktutil 命令将 KDC krb5.keytab.audit 文件与 arstore 的 /etc/krb5/krb5.keytab 文件中的现有密钥进行合并。

最后，管理员在被审计系统上生成密钥。

```
enigma # kclient

.. Enter the Kerberos realm: EXAMPLE.COM

.. KDC hostname for the above realm: kserv.example.com

.. Will this client need service keys ? [y/n]: y
```

▼ 如何配置 syslog 审计日志

可以指示审计服务，将审计队列中的部分或全部审计记录复制到 syslog 实用程序。如果记录二进制审计数据和文本摘要，二进制数据将提供完整的审计记录，而摘要将筛选数据以供实时查看。

开始之前 要配置 audit_syslog 插件，您必须是指定有 "Audit Configuration"（审计配置）权限配置文件的管理员。要配置 syslog 实用程序并创建 auditlog 文件，您必须承担 root 角色。

1. 选择要发送到 **audit_syslog** 插件的审计类，并激活该插件。

   注 - p_flags 审计类必须预选为系统缺省值，或在用户的审计标志或权限配置文件的审计标志中预选。不会为未预选的类收集记录。

   ```
   # auditconfig -setplugin audit_syslog \
        active p_flags=lo,+as,-ss
   ```

2. 配置 **syslog** 实用程序。

   a. 将 **audit.notice** 项添加到 **syslog.conf** 文件中。
      此项包括日志文件的位置。

      ```
      # cat /etc/syslog.conf

      …
      audit.notice       /var/adm/auditlog
      ```

   b. 创建日志文件。

      ```
      # touch /var/adm/auditlog
      ```

   c. 将日志文件的权限设置为 640。

      ```
      # chmod 640 /var/adm/auditlog
      ```

   d. 检查系统上运行了哪些系统日志服务实例。

```
# svcs system-log

STATE        STIME      FMRI
online       Nov_27     svc:/system/system-log:default
disabled     Nov 27     svc:/system/system-log:rsyslog
```

e. 刷新活动 **syslog** 服务实例的配置信息。

```
# svcadm refresh system/system-log:default
```

3. 刷新审计服务。

审计服务在刷新时将读取审计插件的更改。

```
# audit -s
```

4. 定期归档 **syslog** 日志文件。

审计服务可生成大量输出。要管理日志，请参见 logadm(1M) 手册页。

例 4-11 指定 syslog 输出的审计类

在以下示例中，syslog 实用程序收集预选的审计类的子集。pf 类是在例 3-15 “创建新的审计类” 中创建的。

```
# auditconfig -setnaflags lo,na

# auditconfig -setflags lo,ss

# usermod -K audit_flags=pf:no jdoe

# auditconfig -setplugin audit_syslog \
    active p_flags=lo,+na,-ss,+pf
```

auditconfig 命令的参数指示系统收集所有登录/注销、无归属事件以及系统状态审计记录的更改。audit_syslog 插件项指示 syslog 实用程序收集所有登录、成功的无归属事件以及失败的系统状态更改。

对于 jdoe 用户，该二进制实用程序会收集对 pfexec 命令的成功和失败调用。syslog 实用程序收集对 pfexec 命令的成功调用。

例 4-12 将 syslog 审计记录放置在远程系统上

可以更改 syslog.conf 文件中的 audit.notice 项，使其指向远程系统。在本示例中，本地系统的名称为 sys1.1。远程系统是 remote1。

```
sys1.1 # cat /etc/syslog.conf

…
audit.notice       @remote1
```

在 remote1 系统上，syslog.conf 文件中的 audit.notice 项指向日志文件。

```
remote1 # cat /etc/syslog.conf

…
audit.notice       /var/adm/auditlog
```

♦ ♦ ♦ 第 5 章

5 使用审计数据

本章提供的过程可帮助您使用不同本地系统生成的审计数据。本章包含以下主题：

此外，以下各章介绍了其他审计管理任务：

- 第 3 章 管理审计服务
- 第 4 章 监视系统活动
- 第 6 章 分析和解决审计服务问题

有关审计服务的概述，请参见第 1 章 关于 Oracle Solaris 中的审计。有关规划建议，请参见第 2 章 规划审计。有关参考信息，请参见第 7 章 审计参考。

显示审计迹数据

缺省插件 `audit_binfile` 创建审计迹。审计迹可能包含大量数据。以下各节介绍如何使用该数据。

显示审计记录定义

要显示审计记录定义，请使用 `auditrecord` 命令。定义提供审计事件的审计事件编号、审计类、选择掩码和记录格式。

```
% auditrecord -options
```

该命令生成的屏幕输出取决于所使用的选项，如以下显示了部分选项的列表所示。

- `-p` 选项显示程序的审计记录定义。
- `-c` 选项显示审计类的审计记录定义。

- -a 选项用于列出所有审计事件定义。

您还可以将显示的输出打印到文件。

有关更多信息，请参见 auditrecord(1M) 手册页。

例 5-1 显示程序的审计记录定义

在本示例中，将显示 login 程序生成的所有审计记录的定义。登录程序包括 rlogin、telnet、newgrp，以及 Oracle Solaris 的 安全 Shell 功能。

```
% auditrecord -p login
...
login: logout
program     various              See login(1)
event ID    6153                 AUE_logout
class       lo                  (0x0000000000001000)
...
newgrp
program     newgrp               See newgrp login
event ID    6212                 AUE_newgrp_login
class       lo                  (0x0000000000001000)
...
rlogin
program     /usr/sbin/login      See login(1) - rlogin
event ID    6155                 AUE_rlogin
class       lo                   (0x0000000000001000)
...
/usr/lib/ssh/sshd
program     /usr/lib/ssh/sshd    See login - ssh
event ID    6172                 AUE_ssh
class       lo                   (0x0000000000001000)
...
telnet login
program     /usr/sbin/login      See login(1) - telnet
event ID    6154                 AUE_telnet
class       lo                   (0x0000000000001000)
…
```

例 5-2 显示审计类的审计记录定义

在本示例中，显示在例 3-15 “创建新的审计类” 中创建的 pf 类中所有审计记录的定义。

```
% auditrecord -c pf
pfexec
system call pfexec               See execve(2) with pfexec enabled
event ID    116                  AUE_PFEXEC
class       pf                   (0x0100000000000000)
header
```

```
path                    pathname of the executable
path                    pathname of working directory
[privileges]            privileges if the limit or inheritable set are changed
[privileges]            privileges if the limit or inheritable set are changed
[process]               process if ruid, euid, rgid or egid is changed
exec_arguments
[exec_environment]      output if arge policy is set
subject
[use_of_privilege]
return
```

使用特权时，将记录 `use_of_privilege` 标记。更改限制或可继承权限集时，将记录 `privileges` 标记。更改 ID 时，将记录 `process` 标记。无需任何策略选项即可将这些标记包含在记录中。

例 5-3 将审计记录定义打印到文件

在本示例中，添加了 `-h` 选项，以便将所有审计记录定义以 HTML 格式放置在一个文件中。在浏览器中显示 HTML 文件时，请使用浏览器的“查找”工具来查找特定审计记录定义。

```
% auditrecord -ah > audit.events.html
```

选择要显示的审计事件

对于指定有 "Audit Review"（审计查看）权限配置文件的管理员，可以使用 `auditreduce` 命令过滤审计记录以进行检查。该命令可以在合并输入文件时删除不太感兴趣的记录。

auditreduce -*option argument* **[***optional-file***]**

其中，*argument* 是选项需要的特定参数。

以下是部分记录选择选项及其对应参数的列表：

`-c` 选择一个审计类，其中 *argument* 是审计类，如 `ua`。

`-d` 选择特定日期的所有事件。*argument* 的日期格式为 *yyymmdd*。其他日期选项（如 `-b` 和 `-a`）可分别选择特定日期之前和之后的事件。

`-u` 选择归属于特定用户的所有事件。您需要为此选项指定一个用户名。另一个用户选项 `-e` 选择归属于有效用户 ID 的所有事件。

`-g` 选择归属于特定组的所有事件。需要为此选项指定一个组名称。

`-c` 选择预选审计类中的所有事件。要使用该选项，请指定一个审计类名称。

-m	选择特定审计事件的所有实例。
-o	按对象类型进行选择。使用此选项可按文件、组、文件所有者、FMRI、PID 和其他对象类型进行选择。
optional-file	审计文件的名称。

该命令也使用全部以大写表示的文件选择选项，如以下示例所示。有关选项的完整列表，请参见 auditreduce(1M) 手册页。

例 5-4 合并和减少审计文件

在本示例中，仅保留审计文件中一个月以前的登录和注销记录。此示例假定当前日期为 9 月 27 日。如果需要获取完整的审计迹，可以从备份介质中恢复。-O 选项将该命令的输出定向到名为 lo.summary 的文件。

```
# cd /var/audit/audit_summary
# auditreduce -O lo.summary -b 20100827 -c lo; compress *lo.summary
```

例 5-5 将一个用户的审计记录复制到摘要文件

在本示例中，合并审计迹中包含特定用户名称的记录。-e 选项用于查找有效用户。-u 选项查找登录用户。-O 选项将输出定向到文件 tamiko。

```
# cd /var/audit/audit_summary
# auditreduce -e tamiko -O tamiko
```

您可以进一步减少显示的信息。在下一个示例中，将过滤以下内容并将其打印到名为 tamikolo 的文件。

- 由 -c 选项指定的用户登录和注销时间。
- 由 -d 选项指定的日期：2013 年 9 月 7 日。日期的简洁形式为 *yyyymmdd*。
- 由 -u 选项指定的用户名：tamiko。
- 由 -M 选项指定的计算机名。

```
# auditreduce -M tamiko -O tamikolo -d 20130907 -u tamiko -c lo
```

例 5-6 将选定记录合并到单个文件中

在本示例中，将从审计迹中选择特定日期的登录和注销记录。将这些记录合并到一个目标文件中。将目标文件写入到包含审计根目录的文件系统以外的文件系统。

```
# auditreduce -c lo -d 20130827 -O /var/audit/audit_summary/logins
```

```
# ls /var/audit/audit_summary/*logins
/var/audit/audit_summary/20130827183936.20130827232326.logins
```

查看二进制审计文件的内容

对于指定有 "Audit Review"（审计查看）权限配置文件的管理员，可以使用 praudit 命令查看二进制审计文件的内容。

```
# praudit options
```

以下是部分选项的列表。可将其中一个选项与 -l 选项组合使用，以便每行显示一个记录。

-s　　以短格式显示审计记录，每行一个标记。

-r　　以原始格式显示审计记录，每行一个标记。

-x　　以 XML 格式显示审计记录，每行一个标记。该选项可用于进一步处理。

通过从 auditreduce 命令传输 praudit 输出，也可以同时使用 auditreduce 和 praudit 命令。

例 5-7　以短格式显示审计记录

在本示例中，将以短格式显示 auditreduce 命令提取的登录和注销事件。

```
# auditreduce -c lo | praudit -s

header,69,2,AUE_screenlock,,mach1,2010-10-14 08:02:56.348 -07:00
subject,jdoe,root,staff,jdoe,staff,856,50036632,82 0 mach1
return,success,0
sequence,1298
```

例 5-8　以原始格式显示审计记录

在本示例中，将以原始格式显示 auditreduce 命令提取的登录和注销事件。

```
# auditreduce -c lo | praudit -r

21,69,2,6222,0x0000,10.132.136.45,1287070091,698391050
36,26700,0,10,26700,10,856,50036632,82 0 10.132.136.45
39,0,0
47,1298
```

例 5-9 以 XML 格式显示审计记录

在本示例中，将审计记录转换为 XML 格式。

```
# praudit -x 20100827183214.20100827215318.logins > 20100827.logins.xml
```

同样，您可以以 XML 格式显示 auditreduce 命令过滤的审计记录。

```
# auditreduce -c lo | praudit -x
<record version="2" event="screenlock - unlock" host="mach1"
iso8601="2010-10-14 08:28:11.698 -07:00">
<subject audit-uid="jdoe" uid="root" gid="staff" ruid="jdoe
rgid="staff" pid="856" sid="50036632" tid="82 0 mach1"/>
<return errval="success" retval="0"/>
<sequence seq-num="1298"/>
</record>
```

可以使用脚本对文件内容进行操作，以提取相关信息。

例 5-10 生成适合在浏览器中阅读的 XML 格式的审计记录

您可以使用 xsltproc 工具将记录重新格式化为 XML 文件，以便能在任何浏览器中浏览这些记录。该工具将样式表定义应用到文件内容。要将重新格式化后的内容放置在单独的文件中，可键入以下命令：

```
# auditreduce -c lo | praudit -x | xsltproc - > logins.html
```

在浏览器中，将按类似于以下所示的格式显示 logins.html 的内容：

```
                    Audit Trail Data

File: time: 2013-11-04 12:54:28.000 -08:00

Event: login - local
time: 2013-11-04 12:54:28.418 -08:00 vers: 2 mod: host: host
SUBJECT audit-uid: jdoe uid: jdoe gid: staff ruid: jdoe rgid: staff
      pid: 1534 sid: 3583012893 tid: 0 0 host
RETURN errval: success retval: 0

Event: connect to RAD
time: 2013-11-04 12:54:52.029 -08:00 vers: 2 mod: host: host
SUBJECT audit-uid: jdoe uid: jdoe gid: staff ruid: jdoe rgid: staff
      pid: 1835 sid: 3583012893 tid: 0 0 host
RETURN errval: success retval: 0

Event: role login
time: 2013-11-08 08:42:52.286 -08:00 vers: 2 mod: host: host
SUBJECT audit-uid: jdoe uid: root gid: root ruid: root rgid: root
      pid: 4265 sid: 3583012893 tid: 0 0 host
RETURN errval: success retval: 0

Event: role logout
```

```
time: 2013-11-08 08:43:37.125 -08:00 vers: 2 mod: host: host
SUBJECT audit-uid: jdoe uid: root gid: root ruid: root rgid: root
      pid: 4265 sid: 3583012893 tid: 0 0 host
RETURN errval: success retval: 0

Event: login - ssh
time: 2013-12-23 12:24:37.292 -08:00 vers: 2 mod: host: host
SUBJECT audit-uid: jsmith uid: jsmith gid: staff ruid: jsmith rgid: staff
      pid: 2002 sid: 39351741 tid: 14632 202240 host.example.com
RETURN errval: success retval: 0

Event: role login
time: 2013-12-23 12:25:07.345 -08:00 vers: 2 mod: fe host: host
SUBJECT audit-uid: jsmith uid: root gid: root ruid: root rgid: root
      pid: 2023 sid: 39351741 tid: 14632 202240 host.example.com
RETURN errval: failure retval: Permission denied

Event: su
time: 2013-12-23 17:19:24.031 -08:00 vers: 2 mod: na host: host
RETURN errval: success retval: 0

Event: su logout
time: 2013-12-23 17:19:24.362 -08:00 vers: 2 mod: na host: host
RETURN errval: success retval: 0

Event: login - ssh
time: 2013-12-23 17:27:21.306 -08:00 vers: 2 mod: host: host
SUBJECT audit-uid: jsmith uid: jsmith gid: staff ruid: jsmith rgid: staff
      pid: 2583 sid: 3401970889 tid: 13861 5632 host.example.com
RETURN errval: success retval: 0

Event: role login
time: 2013-12-23 17:27:28.361 -08:00 vers: 2 mod: host: host
SUBJECT audit-uid: jsmith uid: root gid: root ruid: root rgid: root
      pid: 2593 sid: 3401970889 tid: 13861 5632 host.example.com
RETURN errval: success retval: 0

Event: role logout
time: 2013-12-23 17:30:39.029 -08:00 vers: 2 mod: host: host
SUBJECT audit-uid: jsmith uid: root gid: root ruid: root rgid: root
      pid: 2593 sid: 3401970889 tid: 13861 5632 host.example.com
RETURN errval: success retval: 0
```

Other events

例 5-11 仅显示 pfedit 记录

您可以使用过滤器只提取和查看审计迹中的特定记录。在本示例中，将对用于捕获 pfedit 命令使用情况的记录进行过滤。假设摘要文件为 20130827183936.20130827232326.logins。使用 pfedit 命令时可生成 AUE_admin_edit 事件。因此，要提取 pfedit 记录，应运行以下命令：

```
auditreduce -m AUE_admin_edit 20130827183936.20130827232326.logins | praudit
```

例 5-12 打印整个审计迹

通过打印命令的管道，整个审计迹的输出将传输到打印机。出于安全考虑，打印机的访问权限将受到限制。

```
# auditreduce | praudit | lp -d example.protected.printer
```

例 5-13 查看特定审计文件

在本示例中，将在终端窗口中查看登录文件摘要。

```
# cd /var/audit/audit_summary/logins

# praudit 20100827183936.20100827232326.logins | more
```

例 5-14 使用脚本处理 praudit 输出

您可能需要将 praudit 命令的输出作为多行文本处理。例如，您可能需要选择 auditreduce 命令无法选择的记录。您可以使用简单的 shell 脚本来处理 praudit 命令的输出。下面的简单脚本在每行中放置一条审计记录，搜索用户指定的字符串，然后将审计文件返回到其原始格式。

```
#!/bin/sh
#
## This script takes an argument of a user-specified string.
#  The sed command prefixes the header tokens with Control-A
#  The first tr command puts the audit tokens for one record
#  onto one line while preserving the line breaks as Control-A
#
praudit | sed -e '1,2d' -e '$s/^file.*$//' -e 's/^header/^aheader/' \\
| tr '\\012\\001' '\\002\\012' \\
| grep "$1" \\
```

Finds the user-specified string

```
| tr '\\002' '\\012'
```

Restores the original newline breaks

请注意，脚本中的 ^a 为 Ctrl-A，而不是 ^ 和 a 这两个字符。前缀将 header 标记与可能显示为文本的字符串 header 区分开来。

如果出现类似如下的消息，则表明您没有足够的特权，无法使用 praudit 命令：

```
praudit: Can't assign 20090408164827.20090408171614.sys1.1 to stdin.
```

在配置文件 shell 中运行 praudit 命令。您必须是指定有 "Audit Review"（审计查看）权限配置文件的管理员。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

在本地系统上管理审计记录

以下任务列表列出了有关选择、分析和管理审计记录的过程。

表 5-1 在本地系统上管理审计记录的任务列表

任务	说明	参考
合并审计记录。	将多台计算机中的审计文件合并到一个审计迹中。	如何合并审计迹中的审计文件 [89]
清除错误指定的审计文件。	向审计服务意外打开的审计文件提供结束时间戳。	如何清除 not_terminated 审计文件 [90]
防止审计迹溢出。	防止写满审计文件系统。	“防止审计迹溢出” [92]

▼ 如何合并审计迹中的审计文件

通过合并所有审计目录中的审计文件，可以分析整个审计迹的内容。

注 - 由于审计迹中的时间戳采用的是国际协调时间 (Coordinated Universal Time, UTC)，因此日期和小时必须转换为当前时区才有意义。每当使用标准文件命令而不是 `auditreduce` 命令来处理这些文件时，都要切记这一点。

开始之前 您必须是指定有 "Audit Review"（审计查看）权限配置文件的管理员。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

1. **创建用于存储合并的审计文件的文件系统。**

   为了降低达到磁盘空间上限的可能性，该文件系统应当位于与您在如何为审计文件创建 ZFS 文件系统 [66]中创建的用来存储原始文件的文件系统不同的 *zpool* 中。

2. **合并审计迹中的审计记录。**

   转到用于存储合并的审计文件的目录。从此目录中，将审计记录合并到具有指定后缀的文件中。本地系统上审计迹中的所有目录将被合并，然后放置到此目录中。

   ```
   # cd audit-storage-directory
   # auditreduce -Uppercase-option -O suffix
   ```

   `auditreduce` 命令的大写选项用于处理审计迹中的文件。大写选项包括：

   `-A` 选择审计迹中的所有文件。

   `-C` 只选择完整文件。

-M	选择带特定后缀的文件。后缀可以是机器名，也可以是为摘要文件指定的后缀。
-O	在当前目录中创建一个审计文件，该文件的开始时间和结束时间均为 14 个字符的时间戳且后缀为 *suffix*。
-R *pathname*	指定该选项可读取备用审计根目录 *pathname* 中的审计文件。
-S *server*	指定该选项可读取指定服务器中的审计文件。

有关选项的完整列表，请参见 auditreduce(1M) 手册页。

例 5-15 将审计文件复制到摘要文件

在以下示例中，指定有 "System Administrator"（系统管理员）权限配置文件的管理员将审计迹中的所有文件复制到其他文件系统上的合并文件中。/var/audit/storage 文件系统位于独立于 /var/audit 文件系统（审计根文件系统）的磁盘上。

```
$ cd /var/audit/storage
$ auditreduce -A -O All
$ ls /var/audit/storage/*All
20100827183214.20100827215318.All
```

在以下示例中，仅将完整文件从审计迹复制到合并文件中。完整路径指定为 -O 选项的值。路径的最后组成部分 Complete 用作后缀。

```
$ auditreduce -C -O /var/audit/storage/Complete

$ ls /var/audit/storage/*Complete
20100827183214.20100827214217.Complete
```

在下面的示例中，通过添加 -D 选项删除了原始审计文件。

```
$ auditreduce -C -O daily_sys1.1 -D sys1.1

$ ls *sys1.1
20100827183214.20100827214217.daily_sys1.1
```

▼ 如何清除 not_terminated 审计文件

发生异常系统中断时，如果审计服务的审计文件仍处于打开状态，则审计服务将存在。或者，文件系统无法访问，强制系统切换到新文件系统。在这种情况下，尽管审计文件不再用于审计记录，但此文件仍然保留，并以字符串 not_terminated 作为结束时间戳。使用 auditreduce -O 命令可为此文件提供正确的时间戳。

开始之前 您必须是指定有 "Audit Review"（审计查看）权限配置文件的管理员。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

1. **在审计文件系统上，按创建顺序列出带有 not_terminated 字符串的文件。**

   ```
   # ls -R1t audit-directory */* | grep not_terminated
   ```

   -R 列出子目录中的文件。

   -t 按照从最新到最旧的顺序列出文件。

   -1 将文件列成一列。

2. **清除旧的 not_terminated 文件。**

   对 auditreduce -O 命令指定旧文件的名称。

   ```
   # auditreduce -O system-name old-not-terminated-file
   ```

3. **删除旧的 not_terminated 文件。**

   ```
   # rm system-name old-not-terminated-file
   ```

例 5-16 清除关闭的 not_terminated 审计文件

在以下示例中，查找并重命名 not_terminated 文件，然后删除原始文件。

```
ls -R1t */* | grep not_terminated
…/egret.1/20100908162220.not_terminated.egret
…/egret.1/20100827215359.not_terminated.egret

# cd */egret.1
# auditreduce -O egret 20100908162220.not_terminated.egret
# ls -1t
20100908162220.not_terminated.egret     Current audit file

20100827230920.20100830000909.egret     Cleaned-up audit file

20100827215359.not_terminated.egret     Input (old) audit file

# rm 20100827215359.not_terminated.egret
# ls -1t
20100908162220.not_terminated.egret     Current audit file

20100827230920.20100830000909.egret     Cleaned-up audit file
```

新文件的开始时间戳反映 not_terminated 文件中第一个审计事件的时间。结束时间戳反映此文件中最后一个审计事件的时间。

防止审计迹溢出

如果安全策略要求保存所有审计数据，请遵循以下做法以防止审计记录丢失。

注 - 您必须承担 root 角色。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

- 设置 audit_binfile 插件上的最小空闲大小。

  使用 p_minfree 属性。

  磁盘空间满足最小空闲大小时，audit_warn 电子邮件别名将发送警告。请参见例 4-7 “设置警告的软限制”。
- 设置计划安排以定期归档审计文件。

  通过将文件备份到脱机介质来归档审计文件。此外，还可以将这些文件移动到归档文件系统。

  如果正在使用 syslog 实用程序收集文本审计日志，请归档文本日志。有关更多信息，请参见 logadm(1M) 手册页。
- 设置计划安排以从审计文件系统中删除已归档的审计文件。
- 保存和存储辅助信息。

  归档解释审计记录与审计迹所需的信息。至少应保存 passwd、group 和 hosts 文件。也可以将 audit_event 和 audit_class 文件归档。
- 保留已归档审计文件的记录。
- 正确存储归档介质。
- 减少启用 ZFS 压缩所需的文件系统容量。

  在专用于审计文件的 ZFS 文件系统上，压缩将大大缩小文件。有关示例，请参见如何压缩专用文件系统上的审计文件 [57]。

  另请参见《在 Oracle Solaris 11.2 中管理 ZFS 文件系统 》中的“ZFS 压缩、重复数据删除和加密属性之间的交互”。
- 通过创建摘要文件，减少存储的审计数据量。

  可以使用 auditreduce 命令的选项从审计迹中提取摘要文件。摘要文件只包含指定类型审计事件的记录。要提取摘要文件，请参见例 5-4 “合并和减少审计文件” 和例 5-6 “将选定记录合并到单个文件中”。

♦ ♦ ♦ 第 6 章

分析和解决审计服务问题

本章提供了有助于对审计相关问题进行故障排除的过程。此外，以下各章介绍了其他审计管理任务：

- 第 3 章 管理审计服务
- 第 4 章 监视系统活动
- 第 5 章 使用审计数据

有关审计服务的概述，请参见第 1 章 关于 Oracle Solaris 中的审计。有关规划建议，请参见第 2 章 规划审计。有关参考信息，请参见第 7 章 审计参考。

对审计服务进行故障排除

本节介绍了有助于调试审计问题的各种审计错误消息、首选项以及由其他工具提供的审计。

通常，系统会发送各种通知提醒您审计服务出现错误。如果您认为审计服务存在问题，请查看您的电子邮件和日志文件。

- 读取发送到 `audit_warn` 别名的电子邮件。

  `audit_warn` 脚本将警报消息发送到 `audit_warn` 电子邮件别名。如果缺少正确配置的别名，消息将发送到 `root` 帐户。
- 检查审计服务的日志文件。

  `svcs -s auditd` 命令的输出列出审计服务生成的审计日志的完整路径。
- 检查系统日志文件。

  `audit_warn` 脚本将 `daemon.alert` 消息写入 `/var/log/syslog` 文件。

  `/var/adm/messages` 文件可能会包含信息。

找到并解决问题后，启用或重新启动审计服务。

```
# audit -s
```

以下各节介绍了可能的问题案例及其解决步骤。

注 - 执行任何故障排除任务之前，请确保已获得相应的授权。例如，要配置审计，您必须是指定有 "Audit Configuration"（审计配置）权限配置文件的管理员。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“使用所指定的管理权限”。

未记录审计记录

缺省情况下，审计处于启用状态。如果您认为尚未禁用审计，但没有将任何审计记录发送到活动插件，其原因可能是本节中讨论的下列因素之一或者这些因素的组合。请注意，要修改系统文件，您必须指定有 `solaris.admin.edit/`*path-to-system-file* 授权。缺省情况下，root 角色拥有此授权。

审计服务未运行

要检查审计是否正在运行，请使用以下任意方法：

- 检验当前审计条件。

  以下输出说明审计未在运行中：

  ```
  # auditconfig -getcond
  audit condition = noaudit
  ```

  以下输出说明审计正在运行中：

  ```
  # auditconfig -getcond
  audit condition = auditing
  ```

- 验证审计服务是否正在运行。

  以下输出说明审计未在运行中：

  ```
  # svcs -x auditd

  svc:/system/auditd:default (Solaris audit daemon)
  State: disabled since Sun Oct 10 10:10:10 2010
  Reason: Disabled by an administrator.
  See: http://support.oracle.com/msg/SMF-8000-05
  See: auditd(1M)
  See: audit(1M)
  See: auditconfig(1M)
  See: audit_flags(5)
  See: audit_binfile(5)
  See: audit_syslog(5)
  See: audit_remote(5)
  ```

```
See: /var/svc/log/system-auditd:default.log
Impact: This service is not running.
```

以下输出说明审计服务正在运行中：

```
# svcs auditd
STATE          STIME    FMRI
online         10:10:10 svc:/system/auditd:default
```

如果审计服务没有运行，请启用该服务。有关过程，请参见“启用和禁用审计服务” [37]。

没有处于活动状态的审计插件

可使用以下命令检查是否有任何插件处于活动状态。必须至少有一个插件处于活动状态才能使审计服务起作用。

```
# audit -v
audit: no active plugin found
```

如果没有任何插件处于活动状态，则激活一个插件。

```
# auditconfig -setplugin audit_binfile active
# audit -v
configuration ok
```

未定义审计类

您尝试使用的审计类可能未定义。有关创建 `pf` 类的说明，请参见如何添加审计类 [48]。

例如，以下标志列表包含 Oracle Solaris 软件不提供的 `pf` 类：

```
# auditconfig -getflags
active user default audit flags = pf,lo(0x0100000000000000,00x0100000000001000)
configured user default audit flags = pf,lo(0x0100000000000000,00x0100000000001000)
```

如果不希望定义类，可运行具有有效值的 `auditconfig -setflags` 命令以重置当前标志。否则，请在定义类时确保以下事项：

- 在 `audit_class` 文件中定义了审计类。

  ```
  # grep pf /etc/security/audit_class
  ```

  Verify class exists

  ```
  0x0100000000000000:pf:profile
  ```

- 掩码是唯一的。如果不唯一，请替换该掩码。

```
# grep 0x0100000000000000 /etc/security/audit_class
```
Ensure mask is unique

```
0x0100000000000000:pf:profile
```

没有为审计类指定事件

您正在使用的定制类尽管已经进行了定义，但可能未向它们指定任何事件。

要验证是否已为定制的类指定事件，请使用以下方法之一：

```
# auditconfig -lsevent | egrep " pf|,pf|pf,"
AUE_PFEXEC      116 pf execve(2) with pfexec enabled

# auditrecord -c pf
```
List of audit events assigned to pf class

如果未将事件指定给该类，请为其指定适当的事件。

审计记录的卷过大

在确定在您的站点中必须审计哪些事件后，请按照以下建议来创建仅包含您所需信息的审计文件。请注意，要将标志指定给用户、角色和权限配置文件，您必须承担 `root` 角色。

- 具体而言，应避免向审计迹中添加事件和审计标记。以下策略会增加审计迹的大小。

`arge`	向 `execv` 审计事件添加环境变量。虽然审计 `execv` 事件的开销可能很大，但向审计记录添加变量的开销很小。
argv	向 `execv` 审计事件添加命令参数。向审计记录添加命令参数的开销不大。
group	将组标记添加到包含可选 `newgroups` 标记的审计事件。
path	将 `path` 标记添加到包含可选 `path` 标记的审计事件。
public	如果正在审计文件事件，每次public object（公共对象）发生可审计事件时，都会将一个事件添加到审计迹中。文件类包括 `fa`、`fc`、`fd`、`fm`、`fr`、`fw` 和 `cl`。有关公共文件的定义，请参见“审计术语和概念” [10]。
seq	将序列标记添加到每个审计事件。

trail	将尾部标记添加到每个审计事件。
windata_down	在配置了 Trusted Extensions 的系统上，在有标签窗口中的信息降级时添加事件。
windata_up	在配置了 Trusted Extensions 的系统上，在有标签窗口中的信息升级时添加事件。
zonename	将区域名称添加到每个审计事件。如果全局区域是配置的唯一区域，将字符串 `zone, global` 添加到每个审计事件。

以下审计记录显示了 `ls` 命令的用法。正在审计 `ex` 类，且正在使用缺省策略：

```
header,129,2,AUE_EXECVE,,mach1,2010-10-14 11:39:22.480 -07:00
path,/usr/bin/ls
attribute,100555,root,bin,21,320271,18446744073709551615
subject,jdoe,root,root,root,root,2404,50036632,82 0 mach1
return,success,0
```

以下是启用所有策略时的相同记录：

```
header,1578,2,AUE_EXECVE,,mach1,2010-10-14 11:45:46.658 -07:00
path,/usr/bin/ls
attribute,100555,root,bin,21,320271,18446744073709551615
exec_args,2,ls,/etc/security
exec_env,49,MANPATH=/usr/share/man,USER=jdoe,GDM_KEYBOARD_LAYOUT=us,EDITOR=gedit,
LANG=en_US.UTF-8,GDM_LANG=en_US.UTF-8,PS1=#,GDMSESSION=gnome,SESSIONTYPE=1,SHLVL=2,
HOME=/home/jdoe,LOGNAME=jdoe,G_FILENAME_ENCODING=@locale,UTF-8, PRINTER=example-dbl,
...
path,/lib/ld.so.1
attribute,100755,root,bin,21,393073,18446744073709551615
subject,jdoe,root,root,root,root,2424,50036632,82 0 mach1
group,root,other,bin,sys,adm,uucp,mail,tty,lp,nuucp,daemon
return,success,0
zone,global
sequence,197
trailer,1578
```

- 使用 `audit_syslog` 插件将部分审计事件发送到 `syslog`。

  但不将这些审计事件发送到 `audit_binfile` 或 `audit_remote` 插件。只有当不要求保留发送到 `syslog` 日志的审计事件的二进制记录时，才适合使用此方法。
- 设置更少的系统范围审计标志并审计各个用户。

  通过减少系统范围内审计的审计类的数目，来减少对所有用户的审计量。

  使用 `roleadd`、`rolemod`、`useradd` 和 `usermod` 命令的 `audit_flags` 关键字审计特定用户和角色的事件。有关示例，请参见例 4-11 “指定 `syslog` 输出的审计类” 和 `usermod`(1M) 手册页。

使用 profiles 命令的 always_audit 和 never_audit 属性审计特定权限配置文件的事件。有关信息，请参见 profiles(1) 手册页。

注 - 与其他安全属性一样，审计标志会受到搜索顺序的影响。有关更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“所指定权限的搜索顺序”。

- 创建定制审计类。

  可以在您的站点上创建审计类。仅将需要监视的审计事件分配给这些类。有关过程，请参见如何添加审计类 [48]。

注 - 有关修改审计配置文件的效果的信息，请参见“审计配置文件和软件包” [104]。

二进制审计文件大小无限制地增长

指定有 "Audit Review"（审计查看）权限配置文件的管理员可以限制二进制审计文件的大小，以便于归档和搜索。您也可以使用本节中所述的其中一个选项，以原始文件为基础创建更小的二进制文件。

- 使用 p_fsize 属性限制各个二进制审计文件的大小。

  有关 p_fsize 属性的说明，请参见 audit_binfile (5) 手册页的 OBJECT ATTRIBUTES 部分。

  有关示例，请参见例 4-3 “限制 audit_binfile 插件的文件大小”。
- 使用 auditreduce 命令选择记录，并将这些记录写入更小的文件供进一步分析。

  auditreduce -*lowercase* 选项用于查找特定记录。

  auditreduce -*Uppercase* 选项用于将您的选择写入文件。有关更多信息，请参见 auditreduce(1M) 手册页。另请参见“显示审计迹数据” [81]。

不审计来自其他操作系统的登录

Oracle Solaris OS 可以审计所有登录，无论登录源为何。如果不对登录进行审计，则可能未针对可归属事件和无归属这两类事件设置 lo 类。该类可审计登录、注销和屏幕锁定。缺省情况下审计这些类。

注 - 要审计 ssh 登录，您的系统必须正在 Oracle Solaris 中运行 ssh 守护进程。在 Oracle Solaris 系统上，将针对该审计服务修改此守护进程。有关更多信息，请参见《在 Oracle Solaris 11.2 中管理安全 Shell 访问 》中的“安全 Shell 和 OpenSSH 项目”。

例 6-1 确保已对登录进行审计

在本示例中，前两个命令的输出显示尚未针对可归属事件和无归属事件设置 lo 类。后两个命令则将 lo 类设置为针对登录事件启用审计。

```
# auditconfig -getflags
active user default audit flags = as,st(0x20800,0x20800)
configured user default audit flags = as,st(0x20800,0x20800)

# auditconfig -getnaflags
active non-attributable audit flags = na(0x400,0x400)
configured non-attributable audit flags = na(0x400,0x400)

# auditconfig -setflags lo,as,st
user default audit flags = as,lo,st(0x21800,0x21800)

# auditconfig -setnaflags lo,na
non-attributable audit flags = lo,na(0x1400,0x1400)
```

♦ ♦ ♦ 第 7 章

7 审计参考

本章介绍了重要的审计组件，并涵盖了以下主题：

有关审计的概述，请参见第 1 章 关于 Oracle Solaris 中的审计。有关规划建议，请参见第 2 章 规划审计。有关在站点上配置审计的过程，请参见以下各章：

- 第 3 章 管理审计服务
- 第 4 章 监视系统活动
- 第 5 章 使用审计数据
- 第 6 章 分析和解决审计服务问题

审计服务

缺省情况下，启用审计服务 `auditd`。要了解如何启用、刷新或禁用该服务，请参见“启用和禁用审计服务” [37]。

如果没有客户配置，则使用以下缺省配置：

- 审计所有登录事件。

  审计成功和不成功的登录尝试。
- 审计所有用户的登录和注销事件，包括角色承担和屏幕锁定。
- `audit_binfile` 插件处于活动状态。`/var/audit` 目录存储审计记录，不限制审计文件的大小，且队列大小为 100 条记录。
- 设置了 `cnt` 策略。

  审计记录填满可用磁盘空间时，系统将跟踪丢弃的审计记录数。剩余百分之一的可用磁盘空间时，将会发出警告。
- 设置了以下审计队列控制：
  - 生成记录锁定之前审计队列中的最大记录数是 100
  - 阻塞的审计进程解除阻塞之前审计队列中的最小记录数是 10
  - 审计队列的缓冲区大小是 8192 字节
  - 将审计记录写入审计迹的时间间隔是 20 秒

要显示缺省值，请参见“显示审计服务缺省值” [36]。

使用审计服务，可以设置临时值或活动值。这些值可能不同于已配置的值或属性值。

- 刷新或重新启动审计服务时，不会恢复临时值。

  审计策略和审计队列控件均接受临时值。审计标志没有临时值。
- 已配置的值存储为服务的属性值，因此刷新或重新启动审计服务时会恢复这些值。

权限配置文件控制可以管理审计服务的人员。有关更多信息，请参见“用于管理审计的权限配置文件” [103]。

缺省情况下，以相同方式审计所有区域。请参见“审计和 Oracle Solaris 区域” [104]。

审计服务手册页

下表汇总了审计服务的主要管理手册页。

手册页	汇总
`audit(1M)`	此命令用于控制审计服务的操作 `audit -n` 为 `audit_binfile` 插件启动新的审计文件。 `audit -s` 启用并刷新审计。 `audit -t` 禁用审计。 `audit -v` 验证是否至少一个插件处于活动状态。
`audit_binfile (5)`	缺省审计插件，可将审计记录发送到二进制文件。另请参见“审计插件” [105]。

手册页	汇总
audit_remote(5)	此审计插件用于将审计记录发送到远程接收者。
audit_syslog(5)	此审计插件用于将审计记录的文本摘要发送到 syslog 实用程序。
audit_class(4)	该文件包含审计类定义。八个高序位可供客户创建新的审计类使用。有关在系统升级时修改此文件是所产生的影响的更多信息，请参见如何添加审计类 [48]。
audit_event(4)	此文件包含审计事件定义且将事件映射到审计类。可对此映射进行修改。有关在系统升级时修改此文件所产生的影响的更多信息，请参见如何更改审计事件的类成员身份 [49]。
audit_flags(5)	介绍审计类预选的语法、仅选择失败事件或成功事件的前缀，以及修改现有预选的前缀。
audit.log(4)	介绍二进制审计文件的命名、文件的内部结构以及每个审计标记的结构。
audit_warn(1M)	在审计服务在写入审计记录过程中遇到异常情况时，此脚本会向某个电子邮件别名发出通知。您可以针对自己的站点定制此脚本，以便在可能需要手动干预时发出警告，或者可以指定如何自动处理这些情况。
auditconfig(1M)	此命令用于检索和设置审计配置参数。 发出不带选项的这一 auditconfig，以显示可检索和设置的参数列表。
auditrecord(1M)	此命令用于显示 /etc/security/audit_event 文件中的审计事件定义。有关示例输出，请参见“显示审计记录定义” [81]。
auditreduce(1M)	此命令用于后选及合并以二进制格式存储的审计记录。此命令可以合并来自一个或多个输入审计文件的审计记录，这些记录保持二进制格式。 大写选项会影响文件选择。小写选项会影响记录选择。
auditstat(1M)	此命令用于显示内核审计统计信息。例如，此命令可以显示内核审计队列中的记录数、丢弃的记录数以及由于系统调用而由用户进程在内核中生成的审计记录数。
praudit(1M)	此命令可从标准输入读取二进制格式的审计记录，并以可显示的格式显示这些记录。可从 auditreduce 命令或者单个审计文件或审计文件列表中传输输入。对于当前审计文件，还可以使用 tail -0f 命令生成输入。 有关示例输出，请参见“查看二进制审计文件的内容” [85]。
syslog.conf(4)	此文件配置为将审计记录的文本摘要发送到 audit_syslog 插件的 syslog 实用程序。

用于管理审计的权限配置文件

Oracle Solaris 提供了用于配置审计服务、启用和禁用服务以及分析审计迹的权限配置文件。必须拥有 root 特权才能编辑审计配置文件。

- "Audit Configuration"（审计配置） – 使管理员可以配置审计服务的参数并运行 auditconfig 命令。
- "Audit Control"（审计控制） – 使管理员可以启动、刷新和禁用审计服务，并运行 audit 命令以启动、刷新或停止该服务。
- "Audit Review"（审计查看） – 使管理员可以分析审计记录。此权限配置文件可授权使用 praudit 和 auditreduce 命令读取审计记录。该管理员还可以运行 auditstat 命令。

- "System Administrator"（系统管理员） – 包括 "Audit Review"（审计查看）权限配置文件。拥有 "System Administrator"（系统管理员）权限配置文件的管理员可以分析审计记录。

要配置用于处理审计服务的角色，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“创建角色”。

审计和 Oracle Solaris 区域

非全局区域可以像全局区域一样进行审计，可以设置自己的标志、存储和审计策略。

当以相同方式审计所有区域时，全局区域中的 `audit_class` 和 `audit_event` 文件会为每个区域中的审计提供类到事件的映射。对于按区域名称后选记录，`+zonename` 策略选项非常有用。

区域也可以分别审计。在全局区域设置策略选项 `perzone` 时，每个非全局区域都会运行自己的审计服务，处理自己的审计队列，指定其审计记录的内容和位置。一个非全局区域也可以设置多个审计策略选项。它不能设置影响整个系统的策略，所以非全局区域无法设置 `ahlt` 或 `perzone` 策略。有关更多说明，请参见“装有 Oracle Solaris 区域的系统上的审计” [22]和“规划区域中的审计” [24]。

要了解有关区域的信息，请参见《Oracle Solaris Zones 介绍 》。

审计配置文件和软件包

Oracle Solaris 中的审计配置文件在软件包中标记有 `preserve=renamenew` 软件包属性。设置该属性后，在更新后也将保留您对文件所做的任何修改。有关 `preserve` 值的影响的信息，请参见 `pkg`(5) 手册页。

这些配置文件还标记有 `overlay=allow` 软件包属性。该属性允许您创建自己的包含这些文件的软件包，并用您的软件包中的文件替换 Oracle Solaris 文件。当您将软件包中的 `overlay` 属性设为 `true` 时，`pkg` 子命令（例如 `verify`、`fix`、`revert`，等等）会将结果返回到您的软件包。有关更多信息，请参见 `pkg`(1) 和 `pkg`(5) 手册页。

审计类

Oracle Solaris 定义了作为容纳大量审计事件的方便容器的审计类。

您可以重新配置审计类，也可以创建新的审计类。审计类名称的长度最多为 8 个字符。类描述最多可包含 72 个字符。允许使用数字和非字母数字字符。有关更多信息，请参见 audit_class(4) 手册页和如何添加审计类 [48]。

注意 - all 类可以生成大量数据并快速填满磁盘。仅当有特殊理由需审计所有活动时，才使用 all 类。

审计类语法

可以针对成功、失败或两者对审计类中的事件进行审计。

- 如果不带前缀，则同时针对成功和失败两种情况对事件类进行审计。
- 如果带有加号 (+) 前缀，则仅针对成功情况对事件类进行审计。
- 如果带有减号 (-) 前缀，则仅针对失败情况对事件类进行审计。
- 要修改当前预选，请在前缀或审计标志之前添加插入记号 (^)。例如：
  - 如果为系统预选了 ot，而用户的预选为 ^ot，则不会针对 other 类中的事件审计该用户。
  - 如果为系统预选了 +ot，而用户的预选为 ^+ot，则不会针对 other 类中的成功事件审计该用户。
  - 如果为系统预选了 -ot，而用户的预选为 ^-ot，则不会针对 other 类中的失败事件审计该用户。

要查看审计类预选的语法，请参见 audit_flags(5) 手册页。

可以在以下命令中指定审计类及其前缀：

- 作为 auditconfig 命令选项 -setflags 和 -setnaflags 的参数。
- 作为 audit_syslog 插件的 p_flags 属性的值。您可以指定该属性作为 auditconfig -setplugin audit_syslog active 命令的一个选项。
- 作为 useradd、usermod、roleadd 和 rolemod 命令的 -K audit_flags=*always-audit-flags*:*never-audit-flags* 选项的值。
- 作为 profiles 命令的 -always_audit 和 -never_audit 属性的值。

审计插件

审计插件指定如何处理审计队列中的审计记录。可以按名称指定审计插件：audit_binfile、audit_remote 和 audit_syslog，可用作 auditconfig -setplugin 命令的参数。可通过以下属性进一步指定插件：

- audit_binfile 插件

p_dir 属性 – 将二进制数据发送到的位置

p_minfree 属性 – 在向管理员发出警告之前，磁盘上剩余的最小空间量。

p_fsize 属性 – 审计文件的最大大小。

- audit_remote 插件

  p_hosts 属性 – 要将二进制审计数据发送到的远程验证审计服务器。

  p_retries 属性 – 尝试访问远程验证审计服务器的次数。

  p_timeout 属性 – 两次尝试访问远程验证审计服务器之间的秒数。

- audit_syslog 插件

  p_flags 属性 – 选择要发送到 syslog 的审计记录的文本摘要

- 对于所有插件，插件排队的审计记录的最大数量 – qsize 属性

请参阅 audit_binfile (5)、audit_remote(5)、audit_syslog(5) 以及 auditconfig(1M) 手册页。

审计远程服务器

审计远程服务器 (Audit Remote Server, ARS) 通过一个安全的链路从被审计系统接收审计记录并存储这些记录。

接收依赖于下列配置：

- 具有特定的审计主体和一种 GSS-API 机制的 Kerberos 领域
- 包含至少一个已配置的且处于活动状态的连接组的 ARS
- 连接组中至少有一个被审计系统，并且必须有一个已配置的且处于活动状态的 audit_remote 插件

连接组是在 ARS 的 group 属性中指定的。对于文件管理，group 可以限制审计文件的大小并指定最小空闲空间。指定不同连接组的主要原因是为了在 ARS 上指定不同的存储位置，如例 4-9 “将审计记录以流方式传输到同一 ARS 上的不同文件位置”中所示。

有关 ARS 的更多信息，请参见 ars(5) 手册页。有关 ARS 配置信息，请参见 auditconfig(1M) 手册页中的 -setremote 选项。

要配置被审计系统，请参见 audit_remote(5) 手册页，以及auditconfig(1M) 手册页中的 -setplugin 选项。

审计策略

审计策略可确定是否将其他信息添加到审计迹中。

以下策略向审计记录添加标记：arge、argv、group、path、seq、trail、windata_down、windata_up 和 zonename。Oracle Solaris 的 Trusted Extensions 功能使用 windata_down 和 windata_up 策略。有关更多信息，请参见《Trusted Extensions 配置和管理 》中的第 22 章 “Trusted Extensions 和审计”。

其余的策略则不添加标记。public 策略限制对公共文件的审计。perzone 策略针对非全局区域建立单独的审计队列。ahlt 和 cnt 策略确定无法传送审计记录时所发生的情况。有关详细信息，请参见“同步事件和异步事件的审计策略” [107]。

“了解审计策略” [29]中介绍了不同审计策略选项的影响。有关审计策略选项的说明，请参见 auditconfig(1M) 手册页中的 -setpolicy 选项。要获得可用策略选项的列表，请运行 auditconfig -lspolicy 命令。要获得当前策略，请运行 auditconfig -getpolicy 命令。

同步事件和异步事件的审计策略

ahlt 策略和 cnt 策略一起控制由于审计队列已满而无法接受更多事件时发生的状况。

注 - 如果至少一个插件的队列可以接受审计记录，则不会触发 cnt 或 ahlt 策略。

---

cnt 和 ahlt 策略独立且相关。结合使用这些策略可以带来以下效果：

- -ahlt +cnt 是随附的缺省策略。通过此缺省值，可以允许处理审计事件，即使无法记录事件也是如此。

  -ahlt 策略指明，如果无法将一个异步事件的审计记录放入内核审计队列，系统将对事件计数并继续处理。在全局区域，as_dropped 计数器会记录该计数。

  +cnt 策略指明，如果一个同步事件已到达却无法被放入内核审计队列，系统将对事件计数并继续处理。区域的 as_dropped 计数器会记录该计数。

  -ahlt +cnt 配置通常在那些即使继续处理会导致审计记录丢失，也必须继续处理的站点上使用。auditstatdrop 字段显示区域中丢弃的审计记录数。
- +ahlt -cnt 策略指明，当异步事件无法添加到内核审计队列时，处理将会停止。

  +ahlt 策略指明，如果一个异步事件的审计记录无法被放入内核审计队列，所有处理都将停止。系统将进入混乱状态。异步事件将不会进入审计队列，而必须通过调用栈上的指针恢复。

  -cnt 策略指明，如果一个同步事件无法被放入内核审计队列，则会阻塞尝试发送事件的线程。该线程将被放入休眠队列，直到审计空间可用。不会保留计数。在审计空间可用之前，程序看上去可能处于挂起状态。

  在每个审计事件记录的重要性高于系统可用性的站点上，通常使用 +ahlt -cnt 配置。auditstat wblk 字段显示线程被阻塞的次数。

但是，如果发生异步事件，系统将进入混乱状态，从而导致故障。可以通过保存的故障转储手动恢复审计事件的内核队列。异步事件将不会进入审计队列，而必须通过调用栈上的指针恢复。

- `-ahlt -cnt` 策略指明，如果异步事件无法被放入内核审计队列，将对事件计数，并继续处理。如果一个同步事件无法被放入内核审计队列，则会阻塞尝试发送事件的线程。该线程将被放入休眠队列，直到审计空间可用。不会保留计数。在审计空间可用之前，程序看上去可能处于挂起状态。

  在那些所有同步审计事件的记录比部分异步审计记录的潜在损失重要的站点上，通常使用 `-ahlt -cnt` 配置。`auditstat wblk` 字段显示线程被阻塞的次数。
- `+ahlt +cnt` 策略指明，如果异步事件无法放入内核审计队列，系统将进入混乱状态。如果无法将同步事件放入内核审计队列，系统将对事件计数，然后继续处理。

进程审计特征

以下审计特征在初始登录时设置：

- **进程预选掩码** – 系统范围审计掩码和用户特定审计掩码的组合（如果指定了用户审计掩码）。用户登录时，登录过程将合并预选类，以便为用户进程建立进程预选掩码。进程预选掩码指定生成审计记录的事件。

  以下算法介绍了系统如何获取用户的进程预选掩码：

  `(system-wide default flags` + *always-audit-classes*`)` - *never-audit-classes*

  将 `auditconfig -getflags` 命令结果中的系统范围审计类添加到用户的 *always_audit* 关键字值 `always-audit-classes` 中的类。然后，从总数中减去用户的 *never-audit-classes* 中的类。另请参见 `audit_flags(5)` 手册页。
- **审计用户 ID** – 用户登录时，进程便会获取不变的审计用户 ID。由用户初始进程启动的所有子进程都会继承此 ID。审计用户 ID 有助于履行职责。即使在用户承担角色后，审计用户 ID 仍保持不变。保存在每个审计记录中的审计用户 ID 使您能够始终跟踪追溯到登录用户的操作。
- **审计会话 ID** – 审计会话 ID 在登录时指定。此 ID 将由所有子进程继承。
- **终端 ID** – 对于本地登录，终端 ID 包含本地系统的 IP 地址，后跟标识用户登录的物理设备的唯一数字。通常是通过控制台登录。对应于控制台设备的编号为 `0,0`。对于远程登录，终端 ID 包含远程主机的 IP 地址，后跟远程端口号和本地端口号。

审计迹

审计迹包含二进制审计文件。该迹由 `audit_binfile` 插件创建。审计服务在审计队列中收集记录，并将它们发送到插件，插件将它们写入磁盘。

二进制审计文件名称约定

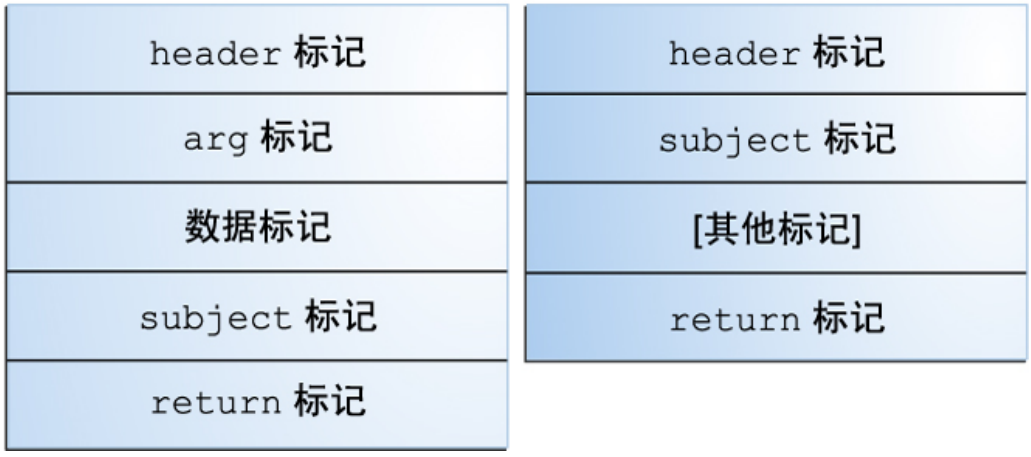
audit_binfile 插件可创建二进制审计文件。每个二进制审计文件都是自包含的记录集合。文件的名称标识生成记录的时间跨度以及生成这些记录的系统。指示时间范围的时间戳是以国际协调时间 (Coordinated Universal Time, UTC) 指定的，以确保它们即使跨越时区也能够以正确的顺序排列。

有关更多信息，请参见 audit.log(4) 手册页。有关打开和关闭的审计文件名称的示例，请参见如何清除 not_terminated 审计文件 [90]。

审计记录结构

审计记录是一系列的审计标记。每个审计标记都包含用户 ID、时间和日期等事件信息。审计记录以 header 标记，以可选的 trailer 标记结束。其他审计标记包含与审计事件相关的信息。下图显示了典型内核审计记录和典型用户级审计记录。

图 7-1 典型的审计记录结构



审计记录分析

审计记录分析涉及从审计迹中后选记录。可以使用两种方法之一来解析收集的二进制数据。

- 您可以使用 praudit 命令。该命令的选项提供不同的文本输出。例如，praudit -x 命令可以为脚本和浏览器中的输入提供 XML。praudit 输出不包括仅用于帮助解析二进制数据的字段。请注意，无法保证 Oracle Solaris 发行版之间 praudit 输出的顺序和格式完全相同。

  有关 praudit 输出的示例，请参见“查看二进制审计文件的内容” [85]。

  有关每个审计标记的 praudit 输出的示例，请参见“审计标记格式” [110]中的各个标记。
- 您可以编写程序来解析二进制数据流。此程序必须考虑审计记录的变体。例如，ioctl（） 系统调用会创建 "Bad file name"（错误的文件名称）的审计记录。该记录包含 "Invalid file descriptor"（无效的文件说明符）的 ioctl（） 审计记录的不同标记。
  - 有关每个审计标记中的二进制数据顺序的说明，请参见 audit.log(4) 手册页。
  - 有关清单值，请参见 /usr/include/bsm/audit.h 文件。
  - 要查看审计记录中的标记顺序，请使用 auditrecord 命令。auditrecord 命令的输出包括不同清单值的不同标记。方括号 ([]) 表明，审计标记是可选的。有关更多信息，请参见 auditrecord(1M) 手册页。

审计标记格式

每个审计标记都有一个标记类型标识符，后跟标记的特定数据。下表显示了每个标记的标记名称以及简短说明。为了与先前的 Solaris 发行版兼容，将维护过时的标记。

表 7-1 用于审计的审计标记

标记名称	说明	详细信息
acl	访问控制条目 (Access Control Entry, ACE) 和访问控制列表 (Access Control List, ACL) 信息	“acl 标记” [112]
arbitrary	具有格式和类型信息的数据	audit.log(4) 手册页
argument	系统调用参数值	“argument 标记” [112]
attribute	文件 vnode 信息	“attribute 标记” [112]
cmd	命令参数和环境变量	“cmd 标记” [112]
exec_args	可执行的系统调用参数	“exec_args 标记” [113]
exec_env	可执行的系统调用环境变量	“exec_env 标记” [113]
exit	程序退出信息	audit.log(4) 手册页
file	审计文件信息	“file 标记” [113]
fmri	框架管理资源指示器	“fmri 标记” [114]
group	进程组信息	“group 标记” [114]
header	表明审计记录开始	“header 标记” [114]

标记名称	说明	详细信息
ip	IP 数据包头信息	audit.log(4) 手册页
ip address	Internet 地址	“ip address 标记” [115]
ip port	Internet 端口地址	“ip port 标记” [115]
ipc	System V IPC 信息	“ipc 标记” [115]
IPC_perm	System V IPC 对象访问信息	“IPC_perm 标记” [116]
opaque	非结构化数据（未指定的格式）	audit.log(4) 手册页
path	路径信息	“path 标记” [116]
path_attr	访问路径信息	“path_attr 标记” [116]
privilege	特权集信息	“privilege 标记” [117]
process	进程信息	“process 标记” [117]
return	系统调用的状态	“return 标记” [117]
sequence	序列号	“sequence 标记” [117]
socket	套接字类型和地址	“socket 标记” [118]
subject	主题信息（格式与 process 相同）	“subject 标记” [118]
text	ASCII 字符串	“text 标记” [118]
trailer	指示审计记录的结束	“trailer 标记” [119]
use of authorization	授权使用情况	“use of authorization 标记” [119]
use of privilege	特权使用情况	“use of privilege 标记” [119]
user	用户 ID 和用户名	“user 标记” [119]
xclient	X 客户机标识	“xclient 标记” [120]
zonename	区域名称	“zonename 标记” [120]
Trusted Extensions 标记	label 和 X 窗口系统信息	《Trusted Extensions 配置和管理 》中的“Trusted Extensions 审计参考”

以下标记已过时：

- liaison
- host
- tid

有关过时标记的信息，请参见包括标记的发行版参考资料。

审计记录总是以 header 标记开头，表明审计记录从审计迹的何处开始。对于可归属事件，subject 和 process 标记引用导致事件发生的进程的值。对于无归属事件，process 标记引用系统。

acl 标记

acl 标记使用两种不同的格式来记录有关适用于 ZFS 文件系统的访问控制条目 (Access Control Entry, ACE) 和适用于 UFS 文件系统的访问控制列表 (Access Control List, ACL) 的信息。

记录 UFS 文件系统的 acl 标记时，praudit -x 命令将显示如下字段：

```
<acl type="1" value="root" mode="6"/>
```

记录 ZFS 数据集的 acl 标记时，praudit -x 命令将显示如下字段：

```
<acl who="root" access_mask="default" flags="-i,-R" type="2"/>
```

argument 标记

argument 标记包含有关系统调用参数的信息：系统调用的参数号、参数值以及可选说明。该标记允许在审计记录中使用 32 位整数的系统调用参数。

praudit -x 命令按如下方式显示 argument 标记的字段：

```
<argument arg-num="2" value="0x5401" desc="cmd"/>
```

attribute 标记

attribute 标记包含文件 vnode 中的信息。

attribute 标记通常与 path 标记同时出现。attribute 标记是在搜索路径期间生成的。如果出现路径搜索错误，则说明没有可用的 vnode 来获取必需的文件信息。因此，attribute 标记不包括在审计记录中。praudit -x 命令按如下方式显示 attribute 标记的字段：

```
<attribute mode="20620" uid="root" gid="tty" fsid="0" nodeid="9267" device="108233"/>
```

cmd 标记

cmd 标记记录与命令关联的参数列表和环境变量的列表。

praudit -x 命令显示 cmd 标记的字段。下面是一个截断的 cmd 标记示例。由于显示的原因进行了换行。

```
<cmd><arge>WINDOWID=6823679</arge>
<arge>COLORTERM=gnome-terminal</arge>
<arge>...LANG=C</arge>...<arge>HOST=machine1</arge>
<arge>LPDEST=printer1</arge>...</cmd>
```

exec_args 标记

exec_args 标记记录 exec（） 系统调用的参数。

praudit -x 命令按如下方式显示 exec_args 标记的字段：

```
<exec_args><arg>/usr/bin/sh</arg><arg>/usr/bin/hostname</arg></exec_args>
```

注 - 仅当 argv 审计策略选项处于活动状态时，才输出 exec_args 标记。

exec_env 标记

exec_env 标记记录 exec（） 系统调用的当前环境变量。

praudit -x 命令显示 exec_env 标记的字段。根据显示的需要，以下示例中进行了换行。

```
<exec_env><env>_=/usr/bin/hostname</env>
<env>LANG=C</env><env>PATH=/usr/bin</env>
<env>LOGNAME=jdoe</env><env>USER=jdoe</env>
<env>DISPLAY=:0</env><env>SHELL=/bin/csh</env>
<env>HOME=/home/jdoe</env><env>PWD=/home/jdoe</env><env>TZ=US/Pacific</env>
</exec_env>
```

注 - 仅当 arge 审计策略选项处于活动状态时，才输出 exec_env 标记。

file 标记

file 标记是一种特殊标记，它可在停用旧文件时标记新审计文件的开始以及旧审计文件的结束。初始 file 标记识别审计迹中的上一个文件。最终 file 标记识别审计迹中的下一个文件。这些标记将连续的审计文件链接成一个审计迹。

praudit -x 命令显示 file 标记的字段。根据显示的需要，以下示例中进行了换行。

```
<file iso8601="2009-04-08 14:18:26.200 -07:00">
```

```
/var/audit/machine1/files/20090408211826.not_terminated.machine1</file>
```

fmri 标记

fmri 标记记录了故障管理资源指示器 (fault management resource indicator, FMRI) 的用法。有关更多信息，请参见 smf(5) 手册页。

praudit -x 命令按如下方式显示 fmri 标记的内容：

```
<fmri service_instance="svc:/system/cryptosvc"</fmri>
```

group 标记

group 标记记录进程凭证中的组项。仅当 group 审计策略选项处于活动状态时，才输出 group 标记。

praudit -x 命令按如下方式显示 group 标记的字段：

```
<group><gid>staff</gid><gid>other</gid></group>
```

header 标记

header 标记的特殊之处在于，它标记审计记录的开始。header 标记与 trailer 标记组合使用，将记录中的所有其他标记括在一起。

在极少的情况下，header 标记可能会包含一个或多个事件修饰符：

- fe 指示失败的审计事件
- fp 指示对特权的使用失败
- na 指示无归属事件

  ```
  header,52,2,system booted,na,mach1,2011-10-10 10:10:20.564 -07:00
  ```

- rd 指示从对象中读取数据
- sp 指示特权的使用成功

  ```
  header,120,2,exit(2),sp,mach1,2011-10-10 10:10:10.853 -07:00
  ```

- wr 指示向对象写入数据

praudit 命令按如下方式显示 header 标记：

```
header,756,2,execve(2),,machine1,2010-10-10 12:11:10.209 -07:00
```

praudit -x 命令在审计记录开始时显示 header 标记的字段。根据显示的需要，以下示例中进行了换行。

```
<record version="2" event="execve(2)" host="machine1"
iso8601="2010-10-10 12:11:10.209 -07:00">
```

ip address 标记

ip address 标记包含 Internet 协议地址（IP 地址）。IP 地址以 IPv4 或 IPv6 格式显示。IPv4 地址使用 4 个字节。IPv6 地址使用 1 个字节来表示地址类型，使用 16 个字节来表示地址。

praudit -x 命令按如下方式显示 ip address 标记的内容：

```
<ip_address>machine1</ip_address>
```

ip port 标记

ip port 标记包含 TCP 或 UDP 端口地址。

praudit 命令按如下方式显示 ip port 标记：

```
ip port,0xf6d6
```

ipc 标记

ipc 标记包含调用者用于标识特殊 IPC 对象的 System V IPC 消息句柄、信号句柄或共享内存句柄。

IPC 对象标识符不符合审计标记的上下文无关性质。没有可唯一标识 IPC 对象的全局名称。IPC 对象由其句柄标识。这些句柄仅当 IPC 对象处于活动状态时才有效。但是，IPC 对象的标识应该不存在问题。很少用到 System V IPC 机制，并且这些机制全部共享相同的审计类。

下表显示了 IPC 对象类型字段的可能值。这些值在 /usr/include/bsm/audit.h 文件中定义。

表 7-2 IPC 对象类型字段的值

名称	值	说明
AU_IPC_MSG	1	IPC 消息对象

名称	值	说明
AU_IPC_SEM	2	IPC 信号对象
AU_IPC_SHM	3	IPC 共享内存对象

praudit -x 命令按如下方式显示 ipc 标记的字段：

```
<IPC ipc-type="shm" ipc-id="15"/>
```

IPC_perm 标记

IPC_perm 标记包含 System V IPC 访问权限的副本。该标记将被添加到由 IPC 共享内存事件、IPC 信号事件和 IPC 消息事件生成的审计记录中。

praudit -x 命令显示 IPC_perm 标记的字段。根据显示的需要，以下示例中进行了换行。

```
<IPC_perm uid="jdoe" gid="staff" creator-uid="jdoe"
creator-gid="staff" mode="100600" seq="0" key="0x0"/>
```

从与 IPC 对象关联的 IPC_perm 结构取得这些值。

path 标记

path 标记包含对象的访问路径信息。

praudit -x 命令按如下方式显示 path 标记的内容：

```
<path>/export/home/srv/.xsession-errors</path>
```

path_attr 标记

path_attr 标记包含对象的访问路径信息。访问路径指定了 path 标记对象下的属性文件对象的顺序。系统调用访问属性文件，如 openat（）。有关属性文件对象的更多信息，请参见 fsattr(5) 手册页。

praudit 命令显示 path_attr 标记如下：

```
path_attr,1,attr_file_name
```

privilege 标记

privilege 标记记录进程中的特权使用情况。不对基本集中的特权记录 privilege 标记。如果已通过管理操作从基本集中删除了特权，则会记录该特权的使用。有关 SMF 的更多信息，请参见《在 Oracle Solaris 11.2 中确保用户和进程的安全 》中的“进程权限管理”。

praudit -x 命令显示 privilege 标记的字段。

```
<privilege set-type="Inheritable">ALL</privilege>
```

process 标记

process 标记包含有关与进程关联的用户（如信号接收者）的信息。

praudit -x 命令显示 process 标记的字段。根据显示的需要，以下示例中进行了换行。

```
<process audit-uid="-2" uid="root" gid="root" ruid="root"
rgid="root" pid="567" sid="0" tid="0 0 0.0.0.0"/>
```

return 标记

return 标记包含系统调用的返回状态 (u_error) 以及进程返回值 (u_rval1)。

return 标记始终作为系统调用的内核生成审计记录的一部分返回。在应用程序审计中，此标记指示退出状态以及其他返回值。

praudit 命令按如下方式显示系统调用的 return 标记：

```
return,failure: Operation now in progress,-1
```

praudit -x 命令按如下方式显示 return 标记的字段：

```
<return errval="failure: Operation now in progress" retval="-1/">
```

sequence 标记

sequence 标记包含一个序列号。每次向审计迹添加一条审计记录，序列号就增加一个数字。仅当 seq 审计策略选项处于活动状态时，才输出 sequence 标记。该标记用于调试。

praudit -x 命令显示 sequence 标记的内容：

```
<sequence seq-num="1292"/>
```

socket 标记

socket 标记包含描述 Internet 套接字的信息。在某些实例中，标记仅包括远程端口和远程 IP 地址。

praudit 命令按如下方式显示该 socket 标记实例：

```
socket,0x0002,0x83b1,localhost
```

展开的标记可添加信息，包括套接字类型和本地端口信息。

praudit -x 命令按如下方式显示该 socket 标记实例。根据显示的需要，以下示例中进行了换行。

```
<socket sock_domain="0x0002" sock_type="0x0002" lport="0x83cf"
laddr="example1" fport="0x2383" faddr="server1.Subdomain.Domain.COM"/>
```

subject 标记

subject 标记描述执行或尝试执行某项操作的用户。格式与 process 标记的格式相同。

subject 标记始终作为系统调用的内核生成审计记录的一部分返回。praudit 命令按如下方式显示 subject 标记：

```
subject,jdoe,root,root,root,root,1631,1421584480,8243 65558 machine1
```

praudit -x 命令显示 subject 标记的字段。根据显示的需要，以下示例中进行了换行。

```
<subject audit-uid="jdoe" uid="root" gid="root" ruid="root"
rgid="root" pid="1631" sid="1421584480" tid="8243 65558 machine1"/>
```

text 标记

text 标记包含一个文本字符串。

praudit -x 命令按如下方式显示 text 标记的内容：

```
<text>booting kernel</text>
```

trailer 标记

header 和 trailer 这两个标记的特殊之处在于，它们将审计记录的各个起止点区分开来，并将所有其他标记括在一起。header 标记指示审计记录的开头。trailer 标记指示审计记录的结尾。trailer 标记是可选标记，仅当已设置 trail 审计策略选项时，才能将此标记添加为每条记录的最后一个标记。

在 trailer 打开的情况下生成审计记录时，auditreduce 命令可以验证 trailer 标记是否正确指回记录 header。trailer 标记支持向后查找审计迹。

praudit 命令按如下方式显示 trailer 标记：

```
trailer,136
```

use of authorization 标记

use of authorization 标记记录授权的使用情况。

praudit 命令按如下方式显示 use of authorization 标记：

```
use of authorization,solaris.role.delegate
```

use of privilege 标记

use of privilege 标记记录特权的使用情况。

praudit -x 命令按如下方式显示 use of privilege 标记的字段：

```
<use_of_privilege result="successful use of priv">proc_setid</use_of_privilege>
```

user 标记

user 标记记录用户名和用户 ID。如果用户名不同于调用者，则出现此标记。

praudit -x 命令按如下方式显示 user 标记的字段：

```
<user uid="123456" username="tester1"/>
```

xclient 标记

xclient 标记包含到 X 服务器的客户机连接数目。

praudit -x 命令按如下方式显示 xlient 标记的内容：

```
<X_client>15</X_client>
```

zonename 标记

zonename 标记记录发生审计事件的区域。字符串 "global" 指示审计事件发生在全局区域。

praudit -x 命令按如下方式显示 zonename 标记的内容：

```
<zone name="graphzone"/>
```

安全词汇表

Access Control List, ACL（访问控制列表）	与传统的 UNIX 文件保护相比，访问控制列表 (access control list, ACL) 可提供更为精细的文件安全性。例如，通过 ACL 可以让组获得对某个文件的读取权限，而仅允许该组中的一个成员获得对该文件的写入权限。
admin principal（管理主体）	名称形式为 *username*`/admin` 的用户主体（如 `jdoe/admin`）。与一般用户主体相比，管理主体可以拥有更多特权（例如，可以更改策略）。另请参见 principal name（主体名称）和 user principal（用户主体）。
AES	Advanced Encryption Standard（高级加密标准）。一种对称的 128 位块数据加密技术。美国政府在 2000 年 10 月采用该种算法的 Rijndael 变体作为其加密标准。AES 从而取代了 user principal（用户主体）加密方法成为政府的加密标准。
algorithm（算法）	加密算法。这是一种确立的递归计算过程，用于对输入执行加密或散列操作。
application server（应用程序服务器）	请参见 network application server（网络应用服务器）。
asynchronous audit event（异步审计事件）	异步事件在系统事件中属于少数。这些事件不与任何进程关联，因此没有任何进程可供阻塞并在以后唤醒。例如，初始系统引导和 PROM 进入和退出事件都是异步事件。
audit files（审计文件）	二进制审计日志。审计文件单独存储在一个审计文件系统中。
audit policy（审计策略）	决定要记录的审计事件的全局设置和按用户设置。通常，应用于审计服务的全局设置会影响审计迹所包括的可选信息。`cnt` 和 `ahlt` 这两个设置会影响系统在填充审计队列时执行的操作。例如，审计策略可能要求每条审计记录都包含一个序列号。
audit trail（审计迹）	来自所有主机的所有审计文件的集合。
authenticated rights profile（已验证的权限配置文件）	要求指定的用户或角色在执行配置文件中的操作之前键入口令的 rights profile（权限配置文件）。该行为与 `sudo` 行为类似。口令的有效时间长度是可配置的。

authentication（验证）	验证主体所声明的身份的过程。
authenticator（验证者）	当客户机从 KDC 请求票证以及从服务器请求服务时，会传递验证者。这些验证者包含使用仅对客户机和服务器公开的会话密钥所生成的信息，这些信息可以作为最新来源进行检验，从而表明事务是安全的。验证者可与票证一起使用来验证用户主体。验证者中包括用户的主体名称、用户主机的 IP 地址，以及时间戳。与票证不同，验证者只能使用一次，通常在请求访问服务时使用。验证者是使用特定客户机和服务器的会话密钥进行加密的。
authorization（授权）	1. 在 Kerberos 中，是指决定主体是否可以使用服务，允许主体访问哪些对象，以及可对每个对象执行的访问操作类型的过程。 2. 在用户权限管理中，是指可以指定给角色或用户（或嵌入权限配置文件中）的权限，此权限用于执行安全策略原本禁止的操作类。在用户应用程序级别（而不是内核）执行操作必须具备授权。
basic set（基本特权集）	登录时为用户进程指定的特权集。在未修改的系统上，每个用户的初始可继承特权集等同于登录时获取的基本特权集。
Blowfish	一种对称块加密算法，它采用 32 位到 448 位的可变长度密钥。其作者 Bruce Schneier 声称 Blowfish 已针对密钥不经常更改的应用程序进行优化。
client principal（客户机主体）	(RPCSEC_GSS API) 是指使用受 RPCSEC_GSS 保护的网络服务的客户机（用户或应用程序）。客户机主体名称将以 `rpc_gss_principal_t` 结构的形式进行存储。
client（客户机）	狭义上讲，是指代表用户使用网络服务的进程，例如，使用 `rlogin` 的应用程序。在某些情况下，服务器本身即可是其他某个服务器或服务的客户机。 广义上讲，是指 a) 接收 Kerberos 凭证的主机，以及 b) 使用由服务器提供的服务的主机。 非正式地讲，是指使用服务的主体。
clock skew（时钟相位差）	所有参与 Kerberos 验证系统的主机上的内部系统时钟可以相差的最大时间量。如果任意两台参与主机之间的时间偏差超过了时钟相位差，则请求会被拒绝。可以在 `krb5.conf` 文件中指定时钟相位差。
confidentiality（保密性）	请参见privacy（保密性）。
consumer（使用者）	在 Oracle Solaris 的加密框架功能中，使用者是指使用提供者提供的加密服务的用户。使用者可以是应用程序、最终用户或内核操作。例如，Kerberos、IKE 和 IPsec 便属于使用者。有关提供者的示例，请参见 provider（提供者）。

credential cache（凭证高速缓存）	包含从 KDC 接收的凭证的存储空间（通常为文件）。
credential（凭证）	包括票证及匹配的会话密钥的信息软件包。用于验证主体的身份。另请参见 ticket（票证）和 session key（会话密钥）。
cryptographic algorithm（密码算法）	请参见 algorithm（算法）。
DES	Data Encryption Standard（数据加密标准）。一种对称密钥加密方法，开发于 1975 年，1981 年由 ANSI 标准化为 ANSI X.3.92。DES 使用 56 位密钥。
device allocation（设备分配）	用户级别的设备保护。设备分配强制规定一次只能由一个用户独占使用一台设备。重用设备之前，将清除设备数据。可以使用授权来限制允许分配设备的用户。
device policy（设备策略）	内核级别的设备保护。设备策略在设备上作为两个特权集实现。一个特权集控制对设备的读取权限，另一个特权集控制对设备的写入权限。另请参见 policy（策略）。
Diffie-Hellman protocol（Diffie-Hellman 协议）	也称为公钥密码学。Diffie 和 Hellman 于 1976 年开发的非对称密钥一致性协议。使用该协议，两个用户可以在以前没有任何密钥的情况下通过不安全的介质交换密钥。Diffie-Hellman 由 Kerberos 使用。
digest（摘要）	请参见 message digest（消息摘要）。
DSA	Digital Signature Algorithm（数字签名算法）。一种公钥算法，采用大小可变（512 位到 4096 位）的密钥。美国政府标准 DSS 可达 1024 位。DSA 的输入依赖于 SHA1。
ECDSA	Elliptic Curve Digital Signature Algorithm（椭圆曲线数字签名算法）。一种基于椭圆曲线数学运算的公钥算法。在生成相同长度的签名时，所需的 ECDSA 密钥大小明显小于 DSA 公钥大小。
effective set（有效特权集）	当前对进程有效的特权集。
flavor（特性）	以前，*security flavor*（安全特性）和 *authentication flavor*（验证特性）具有相同的含义，都是表示验证类型 (AUTH_UNIX, AUTH_DES, AUTH_KERB) 的特性。RPCSEC_GSS 也是一种安全特性，虽然它除了验证之外还提供完整性和保密性服务。
forwardable ticket（可转发票证）	一种票证，可供客户机在不需要完成远程主机上的完整验证过程的情况下用于请求此主机票证。例如，如果用户 `david` 从用户 `jennifer` 的计算机上获取了可转发票证，则 `david` 可以登录到自己的计算机，而不需要获取新票证（因此不需要再次进行自我验证）。另请参见 proxiable ticket（可代理票证）。
FQDN	Fully qualified domain name（全限定域名）。例如，`central.example.com`（与简单的 `denver` 相对）。

GSS-API	Generic Security Service Application Programming Interface（通用安全服务应用编程接口）。为各种模块化安全服务（包括 Kerberos 服务）提供支持的网络层。GSS-API 可用于安全验证服务、完整性服务和保密性服务。另请参见 authentication（验证）、integrity（完整性）和 privacy（保密性）。
hardening（强化）	为了删除主机中固有的安全漏洞而对操作系统的缺省配置进行的修改。
hardware provider（硬件提供者）	在 Oracle Solaris 的加密框架功能中，是指设备驱动程序及其硬件加速器。硬件提供者使计算机系统不必执行开销很大的加密操作，从而可释放 CPU 资源以用于其他用途。另请参见provider（提供者）。
host principal（主机主体）	服务主体的一个特定实例，其中将主体（由主名称 `host` 表示）设置为提供一系列网络服务，如 `ftp`、`rcp` 或 `rlogin`。例如，`host/central.example.com@EXAMPLE.COM` 便是一个主机主体。另请参见 server principal（服务器主体）。
host（主机）	可通过网络进行访问的系统。
inheritable set（可继承特权集）	进程可以通过调用 `exec` 而继承的特权集。
initial ticket（初始票证）	直接颁发（即，不基于现有的票证授予票证）的票证。某些服务（如用于更改口令的应用程序）可能需要将票证标记为 `initial`，以便使其自身确信客户机知晓其密钥。这种保证非常重要，因为初始票证表明客户机最近已进行了自我验证（而并非依赖于存在时间可能较长的票证授予票证）。
instance（实例）	实例是主体名称的第二个部分，用于限定主体的主名称。对于服务主体，实例是必需的。实例就是主机的全限定域名，例如 `host/central.example.com`。对于用户主体，实例是可选的。但是请注意，`jdoe` 和 `jdoe/admin` 都是唯一的主体。另请参见 primary（主）、principal name（主体名称）、service principal（服务主体）和 user principal（用户主体）。
integrity（完整性）	一种安全服务，除了用于用户验证之外，还用于通过加密校验和来验证传输数据的有效性。另请参见 authentication（验证） 和 privacy（保密性）。
invalid ticket（无效票证）	尚未成为可用票证的以后生效的票证。应用服务器将拒绝无效票证，直到此票证生效为止。要使无效票证生效，必须在其开始时间已过后，由客户机通过 TGS 请求将其提供给 KDC，同时设置 `VALIDATE` 标志。另请参见 postdated ticket（以后生效的票证）。
KDC	Key Distribution Center（密钥分发中心）。具有以下三个 Kerberos V5 组件的计算机：

- 主体和密钥数据库
- 验证服务
- 票证授予服务

每个领域都具有一个主 KDC，并且应该具有一个或多个从 KDC。

Kerberos

是指一种验证服务、此服务所使用的协议或者用于实现此服务的代码。

Oracle Solaris 中的 Kerberos 实现主要基于 Kerberos V5 实现。

虽然在技术方面有所不同，但是在 Kerberos 文档中经常会互换使用 "Kerberos" 和 "Kerberos V5"。

Kerberos（也可写成 Cerberus）在希腊神话中是指守护地狱之门的三头凶悍猛犬。

Kerberos policy（Kerberos 策略）

管理 Kerberos 服务中口令的使用的规则集合。这些策略可以控制主体的访问权限或票证参数（如生命周期）。

key（密钥）

1. 通常是指以下两种主要密钥类型之一：

- 对称密钥 – 与解密密钥相同的加密密钥。对称密钥用于对文件进行加密。
- 非对称密钥或公钥 – 在公钥算法（如 Diffie-Hellman 或 RSA）中使用的密钥。公钥包括仅对一个用户公开的私钥、服务器或通用资源所使用的公钥，以及包含这两者的私钥/公钥对。私钥 (private key) 也称为密钥 (secret key)。公钥也称为共享密钥或公用密钥。

2. 密钥表文件中的项（主体名称）。另请参见 keytab file（密钥表文件）。

3. 在 Kerberos 中，是指加密密钥，此类密钥分为以下三种类型：

- 私钥 – 由主体和 KDC 共享并在系统范围之外分发的加密密钥。另请参见 private key（私钥）。
- 服务密钥 – 此密钥与私钥的用途相同，但由服务器和服务使用。另请参见 service key（服务密钥）。
- 会话密钥 – 在两个主体之间使用的临时加密密钥，其生命周期仅限于单个登录会话的持续时间。另请参见 session key（会话密钥）。

keystore（密钥库）

密钥库包含用于应用程序检索的口令、口令短语、证书，以及其他验证对象。密钥库可特定于一种技术，或特定于多个应用程序使用的一个位置。

keytab file（密钥表文件）

包含一个或多个密钥（主体）的密钥表文件。主机或服务使用密钥表文件的方式与用户使用口令的方式大致相同。

kvno	Key version number（密钥版本号）。按照生成顺序跟踪特定密钥的序列号。kvno 最高则表示密钥最新。
least privilege（最小特权）	一种安全模型，该模型仅向指定进程提供超级用户功能的某个子集。最小特权模型为一般用户指定可以用来执行个人管理任务（如挂载文件系统和更改文件的所有权）的足够特权。另一方面，仅使用完成该任务所需的特权运行进程，而不是使用超级用户的完全功能模式（即所有特权）。对非 root 用户而言，可以包含由于编程错误而导致的损坏（如缓冲区溢出），该用户对重要功能（如读取或写入受保护的系统文件或停止计算机）没有访问权限。
limit set（限制特权集）	对哪些特权可用于进程及其子进程的外部限制。
MAC	1. 请参见 message authentication code, MAC（消息验证代码）。 2. 也称为标签设置操作。在政府安全术语中，MAC 是指 Mandatory Access Control（强制访问控制）。例如，Top Secret（绝密）和 Confidential（机密）之类的标签便是 MAC。MAC 与 DAC 相对，后者是指 Discretionary Access Control（自主访问控制）。例如，UNIX 权限便是一个 DAC。 3. 在硬件中，是指 LAN 中的唯一系统地址。如果系统位于以太网中，则 MAC 是指以太网地址。
master KDC（主 KDC）	每个领域中的主要 KDC，包括 Kerberos 管理服务器 `kadmind`，以及验证和票证授予守护进程 `krb5kdc`。每个领域至少都必须具有一个主 KDC，可以具有多个 KDC 副本或从 KDC，这些 KDC 为客户机提供验证服务。
MD5	一种重复加密散列函数，用于进行消息验证（包含数字签名）。该函数于 1991 年由 Rivest 开发。其使用已过时。
mechanism（机制）	1. 指定加密技术以实现数据验证或保密的软件包。例如：Kerberos V5、Diffie-Hellman 公钥。 2. 在 Oracle Solaris 的加密框架功能中，是指用于特殊用途的算法的实现。例如，应用于验证的 DES 机制（如 CKM_DES_MAC）与应用于加密的 DES 机制（如 CKM_DES_CBC_PAD）不同。
message authentication code, MAC（消息验证代码）	MAC 可确保数据的完整性，并验证数据的来源。MAC 不能防止窃听。
message digest（消息摘要）	消息摘要是从消息中计算所得的散列值。此散列值几乎可唯一地标识消息。摘要对检验文件的完整性非常有用。
minimization（最小安装）	运行服务器所需的最小操作系统安装。不安装与服务器操作不直接相关的任何软件，或者在安装之后即删除。
name service scope（名称服务范围）	允许角色在其中执行操作的范围，即，由指定的命名服务（如 NIS 或 LDAP）提供服务的单个主机或所有主机。

network application server（网络应用服务器）
: 提供网络应用的服务器，如 `ftp`。一个领域可以包含多个网络应用服务器。

network policies（网络策略）
: 网络实用程序为了保护网络通信而配置的设置。有关网络安全性的信息，请参见《在 Oracle Solaris 11.2 中确保网络安全 》。

nonattributable audit event（无归属审计事件）
: 无法确定其触发者的审计事件，如 AUE_BOOT 事件。

NTP
: Network Time Protocol（网络时间协议）。由特拉华大学开发的软件，可用于在网络环境中管理准确时间或网络时钟同步，或者同时管理这两者。可以使用 NTP 在 Kerberos 环境中维护时钟相位差。另请参见 clock skew（时钟相位差）。

PAM
: Pluggable Authentication Module（可插拔验证模块）。一种框架，允许使用多种验证机制而不必重新编译运行这些机制的服务。PAM 可用于在登录时初始化 Kerberos 会话。

passphrase（口令短语）
: 一种短语，用于验证某个私钥是否是由口令短语用户创建。理想的口令短语应包含 10-30 个字符，请混合使用字母和数字字符，并且避免简单的文本结构和名称。使用私钥对通信执行加密和解密操作时，系统会提示您提供口令短语进行验证。

password policy（口令策略）
: 可用于生成口令的加密算法，还可以指与口令有关的更普遍的问题，如必须对口令进行更改的频率，允许的口令尝试次数以及其他安全注意事项。安全策略需要口令。口令策略可能要求使用 AES 算法对口令进行加密，并可能对口令强度提出进一步要求。

permitted set（允许特权集）
: 可供进程使用的特权集。

policy for public key technologies（公钥技术的策略）
: 在密钥管理框架 (Key Management Framework, KMF) 中，所实现的策略是管理证书的使用。KMF 策略数据库可以对由 KMF 库管理的密钥和证书的使用施加约束。

policy in the Cryptographic Framework（加密框架中的策略）
: 在 Oracle Solaris 的加密框架功能中，所实现的策略是禁用现有的加密机制。从而使这些机制不可使用。加密框架中的策略可能会阻止使用提供者（如 DES）提供的特殊机制，如 `CKM_DES_CBC`。

policy（策略）
: 一般而言，是指影响或决定决策和操作的操作规划或操作过程。对于计算机系统，策略通常表示安全策略。站点的安全策略是规则集合和相关措施，可用于定义所处理信息的敏感度并防止信息受到未经授权的访问。例如，安全策略可能要求对系统进行审计，设备必须经分配才能使用，以及每六周更改一次口令。

  有关 Oracle Solaris OS 特定区域中的策略实现的信息，请参见 audit policy（审计策略）、policy in the Cryptographic Framework（加密框架中的策略）、device policy（设备策略）、Kerberos policy（Kerberos 策略）、password policy（口令策略）和 rights policy（权限策略）。

postdated ticket（以后生效的票证）

以后生效的票证直到创建之后的某一指定时间才能开始生效。此类票证对于计划在深夜运行的批处理作业等情况非常有用，因为在运行批处理作业之前无法使用该票证（即使被盗）。颁发以后生效的票证时，将以 `invalid` 状态颁发该票证，并在出现以下情况之前一直保持此状态：a) 票证开始时间已过，并且 b) 客户机请求 KDC 进行验证。通常，以后生效的票证在票证授予票证的截止时间之前会一直有效。但是，如果将以后生效的票证标记为 `renewable`，则通常会将其生命周期设置为等于票证授予票证的整个生命周期的持续时间。另请参见 invalid ticket（无效票证）和 renewable ticket（可更新票证）。

primary（主）

主体名称的第一部分。另请参见 instance（实例）、principal name（主体名称）和 Realm（领域）。

principal name（主体名称）

1. 主体的名称，格式为 *primary/instance@REALM*。另请参见 instance（实例）、primary（主）和 Realm（领域）。

2.(RPCSEC_GSS API) 请参见 client principal（客户机主体）和 server principal（服务器主体）。

principal（主体）

1. 参与网络通信并且具有唯一名称的客户机/用户或服务器/服务实例。Kerberos 事务涉及主体之间（服务主体与用户主体）或主体与 KDC 之间的交互。换言之，主体是 Kerberos 可为其指定票证的唯一实体。另请参见 principal name（主体名称）、service principal（服务主体）和 user principal（用户主体）。

2.(RPCSEC_GSS API) 请参见 client principal（客户机主体）和 server principal（服务器主体）。

principle of least privilege（最小特权原则）

请参见 least privilege（最小特权）。

privacy（保密性）

一种安全服务，其中传输的数据加密之后才会发送。保密性还包括数据完整性和用户验证。另请参见 authentication（验证）、integrity（完整性）和 service（服务）。

private key（私钥）

为每个用户主体提供的密钥，并且只对主体的用户和 KDC 公开。对于用户主体，密钥基于用户的口令。另请参见 key（密钥）。

private-key encryption（私钥加密）

采用私钥加密时，发送者和接收者使用相同的加密密钥。另请参见 public-key encryption（公钥加密）。

privilege escalation（特权升级）

获取对未指定权限的资源的访问权限，包括覆盖缺省项的权限和许可。特权升级的结果是某个进程可以执行未经授权的操作。

privilege model（特权模型）

计算机系统上比超级用户模型更为严格的安全模型。在特权模型中，进程需要具有相应的特权才能运行。系统管理可以分为多个独立的部分，这些部分基于管理员在其进程中所具有的特权。可以将特权指定给管理员的登录过程。或者，可以指定特权只对特定命令有效。

privilege set（特权集）

特权的集合。每个进程都有四个特权集，用于确定进程是否可以使用特定特权。请参见 limit set（限制特权集）、effective set（有效特权集）、permitted set（允许特权集）和 inheritable set（可继承特权集）。

此外，特权的 basic set（基本特权集）是指登录时为用户进程指定的特权集合。

privilege-aware（特权识别）

在其代码中启用和禁用特权的程序、脚本和命令。在生产环境中，启用的特权必须提供给进程，例如，通过要求程序的用户使用将特权添加到程序中的权限配置文件。有关特权的完整说明，请参见 `privileges(5)` 手册页。

privilege（特权）

1. 通常是指高于一般用户权限的在计算机系统上执行操作的权限或能力。超级用户特权是授予超级用户的所有 rights（权限）。特权用户或特权应用程序是授予额外权限的用户或应用程序。

2. 是指 Oracle Solaris 系统中的进程所具有的独立权限。与 `root` 相比，特权可提供更为精细的进程控制。特权是在内核中定义和实施的。特权也称为进程特权或内核特权。有关特权的完整说明，请参见 `privileges(5)` 手册页。

privileged application（特权应用程序）

可以覆盖系统控制的应用程序。该应用程序可以检查安全属性（如特定的 UID、GID、授权或特权）。

privileged user（特权用户）

计算机系统上为其指定的权限高于一般用户权限的用户。另请参见 trusted user（信任用户）。

profile shell（配置文件 shell）

在权限管理中，角色（或用户）可在该 shell 中从命令行运行指定给角色权限配置文件的任何特权应用程序。配置文件 shell 版本与系统上可用的 shell 一致，如 `bash` 的 `pfbash` 版本。

provider（提供者）

在 Oracle Solaris 的加密框架功能中，是指为使用者提供的加密服务。例如，PKCS #11 库、内核加密模块和硬件加速器便是提供者。提供者可插入到加密框架中，因此也称为插件。有关使用者的示例，请参见 consumer（使用者）。

proxiable ticket（可代理票证）

可供服务用于代表客户机执行客户机操作的票证。因此，可以说服务充当客户机的代理。使用该票证，服务便可具有客户机的身份。服务可以使用可代理票证来获取其他服务的服务票证，但是不能获取票证授予票证。可代理票证与可转发票证之间的区别在于可代理票证只对单项操作有效。另请参见 forwardable ticket（可转发票证）。

public object（公共对象）

`root` 用户所拥有且全局可读的文件，如 `/etc` 目录中的任何文件。

public-key encryption（公钥加密）

一种加密方案，其中每个用户都有两个密钥：一个是公钥，一个是私钥。采用公钥加密时，发送者使用接收者的公钥对消息进行加密，而

	接收者则使用私钥对其进行解密。Kerberos 服务是一种私钥系统。另请参见 private-key encryption（私钥加密）。
QOP	保护质量。用于选择与完整性服务或保密性服务结合使用的加密算法的参数。
RBAC	基于角色的访问控制，Oracle Solaris 的用户权限管理功能。请参见 rights（权限）。
RBAC policy（RBAC 策略）	请参见 rights policy（权限策略）。
Realm（领域）	1. 由单个 Kerberos 数据库以及一组密钥分发中心 (Key Distribution Center, KDC) 提供服务的逻辑网络。 2. 主体名称的第三部分。对于主体名称 `jdoe/admin@CORP.EXAMPLE.COM`，领域为 `CORP.EXAMPLE.COM`。另请参见 principal name（主体名称）。
reauthentication（重新验证）	需要提供口令才能执行计算机操作。`sudo` 操作通常需要重新验证。已验证的权限配置文件可包含需要重新验证的命令。请参见 authenticated rights profile（已验证的权限配置文件）。
relation（关系）	在 `kdc.conf` 或 `krb5.conf` 文件中定义的配置变量或关系。
renewable ticket（可更新票证）	由于票证的生命周期过长会存在安全风险，因此可以将票证指定为 `renewable`。可更新票证有两个截止时间：a) 票证的当前实例的截止时间，b) 任意票证的最长生命周期。如果客户机需要继续使用某票证，则可在首次失效之前更新此票证。例如，某个票证的有效期为 1 小时，所有票证的最长生命周期为 10 小时。如果持有票证的客户机希望保留此票证的时间长于 1 小时，则必须更新此票证。当某个票证达到最长票证生命周期时，便会自动到期，并且无法更新。
rights policy（权限策略）	与命令关联的安全策略。目前，`solaris` 是 Oracle Solaris 的有效策略。`solaris` 策略可识别特权和扩展的特权策略、授权及 `setuid` 安全属性。
rights profile（权限配置文件）	也称为配置文件。是可指定给角色或用户的安全覆盖项集合。权限配置文件可包括授权、特权、具有安全属性的命令和称为补充配置文件的其他权限配置文件。
rights（权限）	对超级用户模型（管理员对系统要么具有全部控制权要么毫无控制权）的替代。通过用户权限管理和进程权限管理，组织可划分超级用户的特权并将其指定给用户或角色。Oracle Solaris 中权限的实现方式有内核特权、授权和作为特定 UID 或 GID 运行进程的能力。权限可集中在 rights profile（权限配置文件）和 role（角色）中。
role（角色）	一种用于运行特权应用程序的特殊身份，仅指定用户才能承担此身份。

RSA	获取数字签名和公钥密码系统的方法。该方法于 1978 年首次由其开发者 Rivest、Shamir 和 Adleman 介绍。
scan engine（扫描引擎）	第三方应用程序，驻留在外部主机上，可检查文件中是否含有已知病毒。
SEAM	Solaris 系统上的 Kerberos 初始版本的产品名称。该产品基于麻省理工学院开发的 Kerberos V5 技术。SEAM 现称为 Kerberos 服务。它与 MIT 版本仍有细微的差别。
secret key（密钥）	请参见 private key（私钥）。
Secure Shell（安全 Shell）	一种特殊协议，用于在不安全的网络中进行安全远程登录并提供其他安全网络服务。
security attributes（安全属性）	当超级用户以外的用户运行管理命令时，可使此命令成功执行的安全策略覆盖项。在超级用户模型中，`setuid root` 和 `setgid` 程序都是安全属性。将这些属性应用于某命令时，此命令便会成功执行，而与运行它的用户无关。在 privilege model（特权模型）中，内核特权和其他 rights（权限）将 `setuid root` 程序替换为安全属性。特权模型与超级用户模型兼容，因为特权模型也可将 `setuid` 和 `setgid` 程序识别为安全属性。
security flavor（安全特性）	请参见 flavor（特性）。
security mechanism（安全机制）	请参见 mechanism（机制）。
security policy（安全策略）	请参见 policy（策略）。
security service（安全服务）	请参见 service（服务）。
seed（种子）	用于生成随机数的数字起动器。当起动器来自随机源时，种子称为随机种子。
separation of duty（职责分离）	least privilege（最小特权）的部分概念。职责分离可阻止一个用户执行或批准完成事务的所有操作。例如，在RBAC中，可以将登录用户的创建与安全覆盖的指定分隔开来。一个角色创建该用户。另一个角色可以将安全属性（如权限配置文件、角色和特权）指定给现有用户。
server principal（服务器主体）	(RPCSEC_GSS API) 提供服务的主体。服务器主体以 *service@host* 形式的 ASCII 字符串进行存储。另请参见 client principal（客户机主体）。
server（服务器）	为网络客户机提供资源的主体。 例如，如果通过 `ssh` 远程登录到系统 `central.example.com`，则该系统便是提供 `ssh` 服务的服务器。另请参见 service principal（服务主体）。
service key（服务密钥）	由服务主体和 KDC 共享，并在系统范围之外分发的加密密钥。另请参见 key（密钥）。

service principal（服务主体）	为一项或多项服务提供 Kerberos 验证的主体。对于服务主体，主名称是服务的名称（如 `ftp`），其实例是提供服务的系统的全限定主机名。另请参见 host principal（主机主体）和 user principal（用户主体）。
service（服务）	1. 通常由多台服务器提供给网络客户机的资源。例如，如果通过 `rlogin` 远程登录到计算机 `central.example.com`，则该计算机便是提供 `rlogin` 服务的服务器。 2. 除验证之外，还提供其他保护级别的安全服务（完整性或保密性）。另请参见 integrity（完整性）和 privacy（保密性）。
session key（会话密钥）	由验证服务或票证授予服务生成的密钥。生成会话密钥的目的是在客户机与服务之间提供安全事务。会话密钥的生命周期仅限于单个登录会话的持续时间。另请参见 key（密钥）。
SHA1	安全散列算法。该算法可以针对长度小于 2^{64} 的任何输入进行运算，以生成消息摘要。SHA1 算法是 DSA 的输入。
single-system image（单系统映像）	单系统映像用在 Oracle Solaris 审计中来描述使用相同命名服务的一组被审计系统。这些系统将其审计记录发送给某个中心审计服务器，可在该服务器中对记录进行比较，就像这些记录来自一个系统一样。
slave KDC（从 KDC）	主 KDC 的副本，可以执行主 KDC 的大多数功能。每个领域通常都具有若干个从 KDC（但仅有一个主 KDC）。另请参见 KDC 和 master KDC（主 KDC）。
software provider（软件提供者）	在 Oracle Solaris 的加密框架功能中，是指提供加密服务的内核软件模块或 PKCS #11 库。另请参见provider（提供者）。
stash file（存储文件）	存储文件包含 KDC 主密钥的已加密副本。当重新引导服务器以便在 KDC 启动 `kadmind` 和 `krb5kdc` 进程之前自动验证 KDC 时，将使用此主密钥。由于存储文件中包含主密钥，因此，应该保证存储文件及其任何备份的安全。如果加密受到威胁，则可以使用此密钥来访问或修改 KDC 数据库。
superuser model（超级用户模型）	计算机系统上的典型 UNIX 安全模型。在超级用户模型中，管理员对系统要么具有全部的控制权要么毫无控制权。通常，为了管理计算机，用户可成为超级用户 (`root`)，并可执行所有管理活动。
synchronous audit event（同步审计事件）	审计事件中的大多数事件属于同步审计事件。这些事件与系统中的某个进程关联。与某个进程关联的无归属事件属于同步事件，如失败的登录。
TGS	票证授予服务。负责颁发票证的那部分 KDC。
TGT	票证授予票证。由 KDC 颁发的票证，客户机可使用此票证来请求其他服务的票证。
ticket file（票证文件）	请参见 credential cache（凭证高速缓存）。

ticket（票证）	用于安全地将用户身份传递给服务器或服务的信息包。一个票证仅对一台客户机以及某台特定服务器上的一项特殊服务有效。票证包含服务的主体名称、用户的主体名称、用户主机的 IP 地址、时间戳以及定义此票证生命周期的值。票证是通过由客户机和服务使用的随机会话密钥创建的。一旦创建了票证，便可重复使用此票证，直到其到期为止。票证与新的验证者同时出现时，仅用于验证客户机。另请参见 authenticator（验证者）、credential（凭证）、service（服务）和 session key（会话密钥）。
trusted user（信任用户）	您确定的可在某个信任级别执行管理任务的用户。通常情况下，管理员首先为可信用户创建登录，然后指定与用户的信任和能力级别匹配的管理权限。之后，这些用户便能帮助配置和维护系统。也称为特权用户。
user principal（用户主体）	属于某个特定用户的主体。用户主体的主名称是用户名，其可选实例是用于说明相应凭证预期用法的名称（例如 `jdoe` 或 `jdoe/admin`）。也称为用户实例。另请参见 service principal（服务主体）。
virtual private network, VPN（虚拟专用网络）	通过使用加密和隧道连接公共网络上的用户来提供安全通信的网络。

索引

T

Y

Z