



Solaris™ Security Toolkit 4.2

管理指南

Sun Microsystems, Inc.
www.sun.com

文件号码 819-3791-10
2005 年 7 月, 修订版 A

请将有关本文档的意见和建议提交至: <http://www.sun.com/hwdocs/feedback>

版权所有 2005 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. 保留所有权利。

对于本文档中介绍的产品，Sun Microsystems, Inc. 对其所涉及的技术拥有相关的知识产权。需特别指出的是（但不局限于此），这些知识产权可能包含在 <http://www.sun.com/patents> 中列出的一项或多项美国专利，以及在美国和其他国家/地区申请的一项或多项其他专利或待批专利。

本文档及其相关产品的使用、复制、分发和反编译均受许可证限制。未经 Sun 及其许可方（如果有）的事先书面许可，不得以任何形式、任何手段复制本产品或本文档的任何部分。

第三方软件，包括字体技术，均已从 Sun 供应商处获得版权和使用许可。

本产品的某些部分可能是从 Berkeley BSD 系统衍生出来的，并获得了加利福尼亚大学的许可。UNIX 是 X/Open Company, Ltd. 在美国和其他国家/地区独家许可的注册商标。

Sun、Sun Microsystems、Sun 徽标、Sun BluePrints、Solaris、SunOS、Java、JumpStart、Sun4U、SunDocs 和 Solstice DiskSuite 是 Sun Microsystems, Inc. 在美国和其他国家/地区的服务标记、商标或注册商标。

所有 SPARC 商标的使用均已获得许可，它们是 SPARC International, Inc. 在美国和其他国家/地区的商标或注册商标。带有 SPARC 商标的产品均基于由 Sun Microsystems, Inc. 开发的体系结构。ORACLE 是 Oracle Corporation 的注册商标。

OPEN LOOK 和 Sun™ 图形用户界面是 Sun Microsystems, Inc. 为其用户和许可证持有者开发的。Sun 感谢 Xerox 在研究和开发可视或图形用户界面的概念方面为计算机行业所做的开拓性贡献。Sun 已从 Xerox 获得对 Xerox 图形用户界面 (GUI) 的非独占使用许可。该许可也涵盖实施 OPEN LOOK GUI 的 Sun 许可获得者，而其他情况则应符合 Sun 的书面许可协议。

美国政府权利 — 商业用途。政府用户应遵循 Sun Microsystems, Inc. 的标准许可协议，以及 FAR（Federal Acquisition Regulations，即“联邦政府采购法规”）的适用条款及其补充条款。

本文档按“原样”提供，对于所有明示或默示的条件、陈述和担保，包括对适销性、适用性和非侵权性的默示保证，均不承担任何责任，除非此免责声明的适用范围在法律上无效。



目录

前言 xvii

1. 简介 1

使用 Solaris Security Toolkit 软件保护系统安全 1

JumpStart 模式 2

独立模式 2

了解软件组件 3

目录 4

Audit 目录 4

Documentation 目录 4

man 目录 5

Drivers 目录 5

Files 目录 8

Finish 目录 9

OS 目录 9

Packages 目录 10

Patches 目录 10

Profiles 目录 10

Sysidcfg 目录 11

数据信息库 11

维护版本控制	11
配置和定制 Solaris Security Toolkit 软件	12
策略和要求	12
准则	12
2. 保护系统安全：方法论	15
规划与准备	15
考虑风险和收益	15
查看安全策略、标准和相关文档	17
实例 1	17
实例 2	17
确定应用程序和服务要求	18
确定应用程序和可用服务清单	18
确定服务要求	18
开发和执行 Solaris Security Toolkit 配置文件	26
安装软件	26
执行安装前任务	27
备份数据	27
验证系统稳定性	27
执行安装后的任务	28
验证应用程序和服务功能	28
验证安全性配置文件的安装	28
验证应用程序和服务功能	28
维护系统安全	29
3. 升级、安装和运行安全性软件	31
执行规划及安装前的任务	31
软件依赖性	32
确定使用何种模式	32

独立模式	32
JumpStart 模式	33
升级过程	33
▼ 升级 Solaris Security Toolkit 软件和 Solaris 操作系统	34
▼ 仅升级 Solaris Security Toolkit 软件	35
仅升级 Solaris OS	35
下载安全性软件	35
下载 Solaris Security Toolkit 软件	36
▼ 下载 pkg 版本	36
下载推荐的修补程序群集软件	37
▼ 下载推荐的修补程序群集软件	37
下载 FixModes 软件	38
▼ 下载 FixModes 软件	38
下载 OpenSSH 软件	39
▼ 下载 OpenSSH 软件	39
下载 MD5 软件	40
▼ 下载 MD5 软件	40
定制安全性配置文件	42
安装和执行软件	42
在独立模式下执行软件	42
▼ 在独立模式下执行软件	45
审计选项	46
清除选项	46
显示帮助选项	48
驱动程序选项	49
电子邮件通知选项	50
执行历史记录选项	50
最新执行选项	50

输出文件选项	51
抑制显示输出选项	51
根目录选项	52
撤消选项	52
在 JumpStart 模式下执行软件	52
▼ 在 JumpStart 模式下执行软件	53
验证系统修改	53
对服务执行 QA 检查	53
对配置进行安全性评估	54
验证安全性配置文件	55
执行安装后的任务	55
4. 取消系统更改	57
了解如何记录并取消更改	57
撤消系统更改的要求	58
定制脚本以撤消更改	59
检查手动更改的文件	59
使用具有撤消功能的选项	60
备份选项	61
强制选项	61
保留选项	62
输出文件选项	62
抑制显示输出选项	62
电子邮件通知选项	62
撤消系统更改	63
▼ 撤消 Solaris Security Toolkit 运行操作	63
5. 配置和管理 JumpStart 服务器	71
配置 JumpStart 服务器和环境	72

- ▼ 配置 JumpStart 模式 72
- 使用 JumpStart 配置文件模板 74
 - core.profile 74
 - end-user.profile 74
 - developer.profile 75
 - entire-distribution.profile 75
 - oem.profile 75
 - minimal-SunFire_Domain*.profile 75
- 添加和删除客户机 75
 - add-client 脚本 76
 - rm-client 脚本 77
- 6. 审计系统安全性 79**
 - 维护安全性 79
 - 执行加强操作之前检查安全性 80
 - 定制安全性审计 80
 - 审计安全性的准备工作 81
 - 使用选项和控制审计输出 82
 - 命令行选项 82
 - 显示帮助选项 82
 - 电子邮件通知选项 83
 - 输出文件选项 84
 - 抑制显示输出选项 84
 - 详细选项 84
 - 标题与消息输出 85
 - 主机名、脚本名称和时间戳输出 88
 - 进行安全性审计 89
 - ▼ 进行安全性审计 89

7. 保护系统安全	93
规划与准备	93
假设与限制	94
系统环境	94
安全性要求	95
创建安全性配置文件	95
安装软件	95
下载和安装安全性软件	96
▼ 下载和安装安全性软件	96
安装修补程序	96
▼ 安装修补程序	96
指定和安装 OS 群集	97
▼ 指定和安装 OS 群集	97
配置 JumpStart 服务器和客户机	98
准备基础结构	99
▼ 准备基础结构	99
验证和检查 rules 文件	101
定制加强安全性配置	103
启用 FTP 服务	104
▼ 启用 FTP 服务	104
安装 Secure Shell 软件	104
▼ 安装 Secure Shell	105
启用 RPC 服务	105
▼ 启用 RPC	106
定制 syslog.conf 文件	106
▼ 定制 syslog.conf 文件	106
安装客户机	107
▼ 安装客户机	107

质量保证测试 108

▼ 验证配置文件安装 108

▼ 验证应用程序和服务的功能 109

词汇表 111

索引 117

图

- 图 1-1 软件组件结构 3
- 图 1-2 驱动程序控制流程 6

表

表 1-1	定制文件的命名标准	13
表 2-1	列出最近使用的服务	24
表 3-1	将命令行选项与 <code>jass-execute</code> 一起使用	43
表 4-1	将命令行选项与撤消命令一起使用	61
表 5-1	JumpStart <code>add-client</code> 命令	76
表 5-2	JumpStart <code>rm-client</code> 命令	77
表 6-1	将命令行选项与审计命令结合使用	82
表 6-2	审计详细级别	85
表 6-3	在审计输出中显示标题和消息	86
表 6-4	显示主机名、脚本名称和时间戳审计输出	88

代码示例

代码示例 1-1	驱动程序控制流程代码 7
代码示例 2-1	获得有关文件系统对象的信息 19
代码示例 2-3	确定动态加载的应用程序 20
代码示例 2-2	从正在运行的进程中收集信息 20
代码示例 2-4	确定是否正在使用配置文件 21
代码示例 2-5	确定哪些应用程序正在使用 RPC 22
代码示例 2-6	验证 rusers 服务 23
代码示例 2-7	确定使用 RPC 的应用程序的备选方法 24
代码示例 2-8	确定哪些服务或应用程序拥有哪些端口 25
代码示例 2-9	确定哪些进程正在使用文件和端口 25
代码示例 3-1	将一个修补程序文件移动到 /opt/SUNWjass/Patches 目录 38
代码示例 3-2	在独立模式下使用命令行的范例 42
代码示例 3-3	在独立模式下执行软件 45
代码示例 3-4	-c 选项输出范例 47
代码示例 3-5	-h 选项输出范例 48
代码示例 3-6	-d 驱动程序选项输出范例 49
代码示例 3-7	-H 选项输出范例 50
代码示例 3-8	-l 选项输出范例 51
代码示例 3-9	-o 选项输出范例 51
代码示例 3-10	-q 选项输出范例 52

代码示例 4-1	手动更改的文件输出范例	60
代码示例 4-2	可撤销的运行操作的输出范例	64
代码示例 4-3	撤销运行操作处理多个清单文件条目的输出范例	65
代码示例 4-4	撤销异常的输出范例	66
代码示例 4-5	在撤销过程中选择备份选项的输出范例	67
代码示例 4-6	在撤销过程中选择备份选项的输出范例	68
代码示例 6-1	-h 选项输出范例	83
代码示例 6-2	-o 选项输出范例	84
代码示例 6-3	-q 选项输出范例	84
代码示例 6-4	仅报告审计失败的输出范例	86
代码示例 6-5	审计日志条目的输出范例	88
代码示例 6-6	审计运行操作的输出范例	90
代码示例 7-1	在 JumpStart 服务器上添加一台客户机	99
代码示例 7-2	创建一个配置文件	100
代码示例 7-3	修改的脚本的输出范例	100
代码示例 7-4	检查 rules 文件以纠正错误	101
代码示例 7-5	rules 文件的输出范例	102
代码示例 7-6	错误脚本范例	102
代码示例 7-7	正确脚本范例	103
代码示例 7-8	修改的 xsp-firewall-hardening.driver 输出范例	107
代码示例 7-9	评估安全性配置	108

前言

本手册包含的参考信息有助于您了解和使用 Solaris™ Security Toolkit 软件。它主要针对使用 Solaris Security Toolkit 软件来提高 Solaris™ 操作系统 (OS) 版本 8、9 和 10 安全性的读者，例如部署新的 Sun 系统或保护已部署系统安全的管理员、安全顾问以及其他人员。本书中的指导适用于以 JumpStart™ 模式或独立模式来使用该软件。

阅读本书之前

您应当是经过 Sun 认证的 Solaris™ 系统管理员或网络管理员。同时还应该对标准网络协议和拓扑知识有一定的了解。

由于本书所针对的读者在安全性方面的经验或知识千差万别，因此您的经验和知识将决定您应如何使用本书。

本书的结构

您可以将本手册作为用户指南来使用。其中的章节包含使用该软件保护系统安全的信息、指导和准则。本书的结构如下所示：

第 1 章介绍 Solaris Security Toolkit 软件的设计和用途。其中涉及关键组件、功能、优点和所支持平台方面的信息。

第 2 章提供保护系统安全的方法论。在使用 Solaris Security Toolkit 软件保护系统安全之前，您可以应用该软件的过程。

第 3 章提供有关下载、安装和运行 Solaris Security Toolkit 软件以及其他与安全性相关的软件的指导信息。

第 4 章提供有关在加强安全性过程中撤消由 Solaris Security Toolkit 软件所做更改的信息和过程。

第 5 章提供有关配置和管理 JumpStart 服务器以使用 Solaris Security Toolkit 软件的信息。

第 6 章介绍如何使用 Solaris Security Toolkit 软件来审计（验证）系统安全性。可在加强安全性后使用本章的信息和过程维护已建立的安全性配置文件。

第 7 章介绍如何将前面章节中提供的信息和技巧，应用到安装并保护新系统安全的实际情况中。

使用 UNIX 命令

本文档不会介绍基本的 UNIX® 命令和操作过程，如关闭系统、启动系统和配置设备等。欲获知此类信息，请参阅以下文档：

- 系统附带的软件文档
- Solaris 操作系统的有关文档，其 URL 如下：

<http://docs.sun.com>

Shell 提示符

Shell	提示符
C shell	<i>machine-name%</i>
C shell 超级用户	<i>machine-name #</i>
Bourne shell 和 Korn shell	\$
Bourne shell 和 Korn shell 超级用户	#

印刷约定

字体*	含义	示例
AaBbCc123	命令、文件和目录的名称；计算机屏幕输出	编辑 <code>.login</code> 文件。 使用 <code>ls -a</code> 列出所有文件。 % You have mail.
AaBbCc123	用户键入的内容，与计算机屏幕输出的显示不同	% su Password:
<i>AaBbCc123</i>	保留未译的新词或术语以及要强调的词。要使用实名或值替换的命令行变量。	这些称为 <i>class</i> 选项。 要删除文件，请键入 rm filename 。
新词术语强调	新词或术语以及要强调的词。	您 必须 成为超级用户才能执行此操作。
《书名》	书名	阅读《用户指南》的第 6 章。

* 浏览器的设置可能会与这些设置有所不同。

使用硬件型号的通用术语

Sun Fire™ 高端系统指下列型号：

- E25K
- E20K
- 15K
- 12K

Sun Fire 中端系统指下列型号：

- E6900
- E4900
- 6800
- 4810
- 4800
- 3800

Sun Fire 入门级中端系统指下列型号：

- E2900
- Netra 1280
- V1280

- V890
- V880
- V490
- V480

支持的硬件系统

Solaris Security Toolkit 4.2 软件支持运行 Solaris 10 OS 的 SPARC®（仅 64 位）和 x86/x64 系统。Solaris Security Toolkit 4.2 软件还支持运行 Solaris 8 和 9 的 SPARC 32 位系统，例如 Ultra 2 Creator 3D。

支持的 Solaris OS 版本

Sun 对于 Solaris Security Toolkit 软件的支持，仅限于在 Solaris 8、Solaris 9 和 Solaris 10 操作系统中使用该软件。

注 – 对于 Solaris Security Toolkit 4.2 软件，Solaris 10 仅能用于 Sun Fire 高端系统域，而不能用于系统控制器 (SC)。

虽然 Solaris Security Toolkit 4.2 软件能够在 Solaris 2.5.1、Solaris 2.6 和 Solaris 7 操作系统中使用，但 Sun 支持并不涉及在这些操作系统中使用该软件。

Solaris Security Toolkit 软件可自动检测已安装的 Solaris OS 软件的版本，然后运行适合该操作系统版本的任务。

请注意，在本文档提供的实例中，脚本对 OS 版本进行检查时，它检查的是 SunOS™ 版本 5.x，而不是 Solaris OS 版本 2.x、7、8、9 或 10。表 P-1 显示了 SunOS 版本和 Solaris OS 版本之间的相互关系。

表 P-1 SunOS 版本和 Solaris OS 版本之间的相互关系

SunOS 版本	Solaris OS 版本
5.5.1	2.5.1
5.6	2.6
5.7	7

表 P-1 SunOS 版本和 Solaris OS 版本之间的相互关系（续）

SunOS 版本	Solaris OS 版本
5.8	8
5.9	9
5.10	10

支持的 SMS 版本

如果您在 Sun Fire 高端系统中使用 System Management Services (SMS) 来运行系统控制器 (SC)，且使用的 SMS 版本为 1.3、1.4.1 和 1.5，则在所有的 Solaris 8 和 Solaris 9 OS 版本中均支持 Solaris Security Toolkit 4.2。在使用 Solaris Security Toolkit 4.2 软件的 Solaris 10 OS 中，不支持任何版本的 SMS。

注 – 对于 Solaris Security Toolkit 4.2 软件，Solaris 10 仅能用于域，而不能用于系统控制器 (SC)。

相关文档

可通过访问以下网址获取下表中列出的联机文档：

http://www.sun.com/products-n-solutions/hardware/docs/Software/enterprise_computing/systems_management/sst/index.html

应用	书名	文件号码	格式	位置
发行说明	《Solaris Security Toolkit 4.2 发行说明》	819-3798-10	PDF HTML	联机
参考	《Solaris Security Toolkit 4.1 Reference Manual》	819-1503-10	PDF HTML	联机
手册页	《Solaris Security Toolkit 4.1 Man Page Guide》	819-1505-10	PDF	联机

文档、支持和培训

Sun 功能	URL	描述
文档	http://www.sun.com/documentation/	下载 PDF 和 HTML 文档，订购印刷的文档
支持	http://www.sun.com/support/	获取技术支持以及下载修补程序
培训	http://www.sun.com/training/	学习 Sun 课程

第三方 Web 站点

Sun 对本文档提到的第三方 Web 站点的可用性不承担任何责任。对于此类站点或资源中的（或通过它们获得的）任何内容、广告、产品或其他资料，Sun 并不表示认可，也不承担任何责任。对于因使用或依靠此类站点或资源中的（或通过它们获得的）任何内容、产品或服务而造成的或连带产生的实际或名义损坏或损失，Sun 概不负责，也不承担任何责任。

Sun 欢迎您提出意见

Sun 致力于提高其文档的质量，并十分乐意收到您的意见和建议。您可以通过以下网址提交您的意见和建议：

<http://www.sun.com/hwdocs/feedback>

请在您的反馈信息中包含文档的书名和文件号码：

《Solaris Security Toolkit 4.2 管理指南》，文件号码 819-3791-10

第1章

简介

本章介绍 **Solaris Security Toolkit** 软件的设计和用途。其中涉及关键组件、功能、优点和所支持平台方面的信息。本章还提供有关对修改和部署进行版本控制的准则，并提出有关定制 **Solaris Security Toolkit** 软件的重要准则。

本章包含以下主题：

- 第 1 页 “使用 **Solaris Security Toolkit** 软件保护系统安全”
 - 第 3 页 “了解软件组件”
 - 第 11 页 “维护版本控制”
 - 第 12 页 “配置和定制 **Solaris Security Toolkit** 软件”
-

使用 **Solaris Security Toolkit** 软件保护系统安全

Solaris Security Toolkit 软件的非正式名称为 **JumpStart** 体系结构和安全脚本 (**JASS**) 工具包，可提供自动化、可扩展、可伸缩的机制来构建和维护 **Solaris OS** 系统的安全。使用 **Solaris Security Toolkit** 软件可以加强和审计系统的安全性。

以下是在本指南中使用的一些术语，了解它们很重要。

- **加强系统安全性** — 修改 **Solaris OS** 配置以提高系统的安全性。
 - **审计系统安全性** — 确定系统的配置是否与预定义的安全配置文件相符合。
-

注 — 术语**审计系统安全性**指的是 **Solaris Security Toolkit** 软件对安全性状态的自动化验证过程，方法是与预先定义的安全配置文件相比较。本出版物中使用此术语，不表示在使用审计选项后便可保证系统的绝对安全。

- **系统安全性计分** — 计算在审计运行过程中发现的故障的数量。如果未发现（任何种类的）故障，则最终得分为 0。每检测到一个故障，**Solaris Security Toolkit** 便将得分（也称为安全漏洞值）增加 1 分。

可以通过两种模式来安装 **Solaris Security Toolkit** 软件，本节的后面将对此进行简要介绍：

- 第 2 页 “JumpStart 模式”
- 第 2 页 “独立模式”

无论采取哪种安装方式，您都可以使用 **Solaris Security Toolkit** 软件来加强系统安全性并最小化系统。然后，定期使用 **Solaris Security Toolkit** 软件来审计安全系统的配置文件是否被意外或恶意地进行了修改。

JumpStart 模式

系统安装和配置应该尽可能实现自动化（理想状态为 100%）。这包括 OS 安装和配置、网络配置、用户帐户、应用程序和加强系统安全性。**JumpStart** 软件是一种可用于实现 **Solaris OS** 安装自动化的技术。**JumpStart** 软件提供一种通过网络安装系统的机制，很少需要或不需要人为干预。**Solaris Security Toolkit** 软件提供了一个框架和多个脚本，可在基于 **JumpStart** 软件的安装中实现并使大部分与加强 **Solaris OS** 系统安全性相关的任务自动化。**JumpStart Enterprise Toolkit (JET)** 软件可帮助进行基于 **JumpStart** 的安装并包含支持使用 **Solaris Security Toolkit** 加强系统安全性的模块，要获取该软件，请访问 **Sun Software Download** 站点，网址为：

<http://www.sun.com/download/>

有关 **JumpStart** 技术的更多信息，请参阅 **Sun BluePrints™** 书籍《**JumpStart Technology: Effective Use in the Solaris Operating Environment**》。

独立模式

此外，**Solaris Security Toolkit** 软件还具有独立模式。该模式能够执行与 **JumpStart** 模式中相同的所有安全性加强功能，但是只能在经过部署的系统上使用。在上述两种模式中，安全性修改的定制可以并且应该与系统的安全性要求相匹配。

无论采取哪种安装方式，您都可以使用 **Solaris Security Toolkit** 软件来加强系统安全性。然后，定期使用 **Solaris Security Toolkit** 软件来审计安全系统的配置是否被意外或恶意地进行了修改。

了解软件组件

本节概述了 Solaris Security Toolkit 软件的组件结构。Solaris Security Toolkit 软件是多个文件和目录的集合。图 1-1 为该软件的结构示意图。

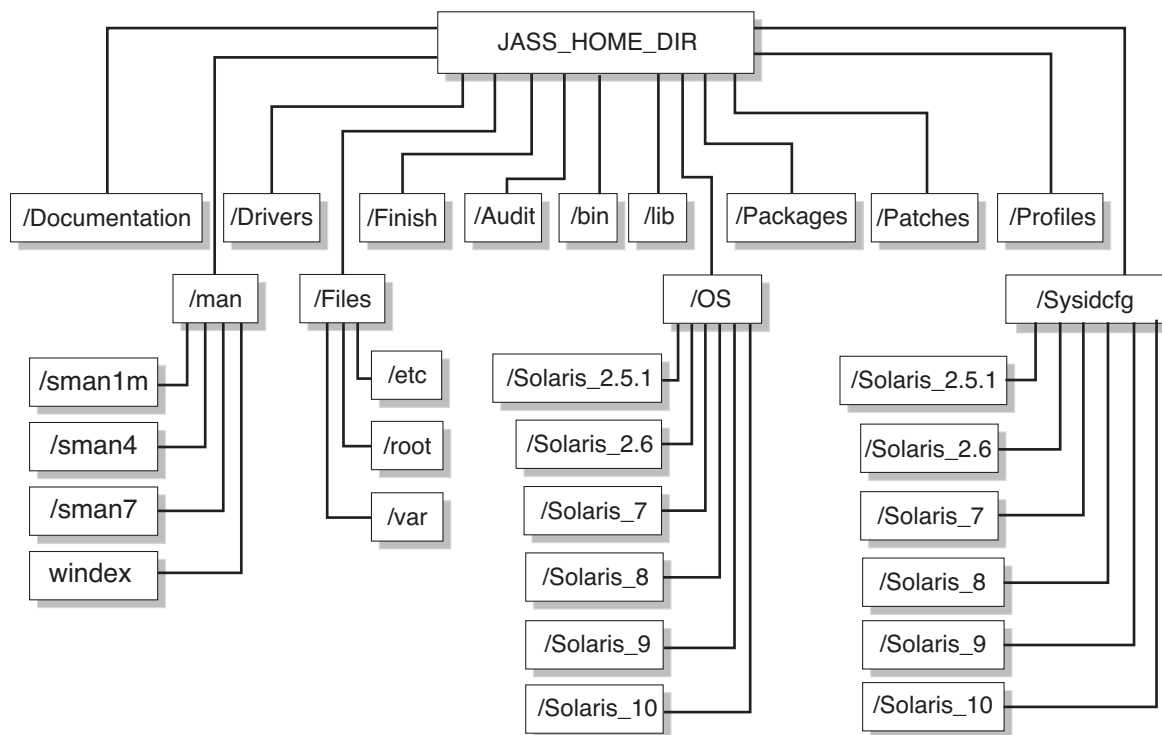


图 1-1 软件组件结构

下列程序或命令文件位于 /bin 目录中：

- add-client — 一个 JumpStart 帮助程序，用于将客户机添加到 JumpStart 环境中
- rm-client — 一个 JumpStart 帮助程序，用于将客户机从 JumpStart 环境中删除
- make-jass-pkg — 该命令能够从 Solaris Security Toolkit 目录内容中创建 Solaris OS 软件包，以简化已定制的 Solaris Security Toolkit 配置在内部的分发
- jass-check-sum — 该命令能够确定曾经由 Solaris Security Toolkit 软件修改过的文件是否已被更改（基于每次 Solaris Security Toolkit 运行过程中所创建的校验和）
- jass-execute — 该命令能够执行 Solaris Security Toolkit 软件的大部分功能

目录

Solaris Security Toolkit 体系结构的组件按以下目录进行组织:

- /Audit
- /bin
- /Documentation
- /Drivers
- /Files
- /Finish
- /lib
- /man
- /OS
- /Packages
- /Patches
- /Profiles
- /Sysidcfg

本节中逐一介绍了每个目录，并在相应的位置列出了每个脚本、配置文件或子目录，同时还提供其他章节作为参考以便您获得详细信息。

Solaris Security Toolkit 目录结构是基于 Sun BluePrints 书籍《JumpStart Technology: Effective Use in the Solaris Operating Environment》中所述的结构。

Audit 目录

该目录包含的脚本，可评估系统是否符合预定义的安全配置文件或审计脚本集。该目录中的脚本分为以下类别：

- 禁用
- 启用
- 安装
- 最小化
- 打印
- 删除
- 设定
- 更新

有关每个类别中脚本的详细列表和每个脚本的说明，请参阅《Solaris Security Toolkit 4.2 Reference Manual》。

Documentation 目录

本目录包含具有用户信息的文本文件，例如 README、EOL_NOTICE 和 INSTALL 文件。

man 目录

本目录包含命令、功能和驱动程序手册页部分的子目录。此外，本目录还包含 `windex` 文件，该文件是命令的索引，且免费提供。

有关这些手册页的更多信息，请参见实际的手册页或《Solaris Security Toolkit 4.2 Man Page Guide》。

Drivers 目录

该目录包含配置信息文件，这些配置信息指定了运行 **Solaris Security Toolkit** 软件时需要执行和安装的文件。该目录还包含驱动程序、脚本和配置文件。

以下是 **Drivers** 目录中的驱动程序和脚本的实例：

- `audit_{private|public}.funcs`
- `common_{log|misc}.funcs`
- `{config|hardening|secure}.driver`
- `driver.{init|run}`
- `driver_{private|public}.funcs`
- `finish.init`
- `server-{config|hardening|secure}.driver`
- `suncluster3x-{config|hardening|secure}.driver`
- `sunfire_15k_sc-{config|hardening|secure}.driver`
- `undo.{funcs|init|run}`
- `user.init.SAMPLE`
- `user.run.SAMPLE`

Solaris Security Toolkit 中的所有驱动程序均具有三个文件：

- 名称-`{config|hardening|secure}.driver`

在以上列表中，这三个文件是在括号中列出的，例如 `sunfire_15k_sc-{config|hardening|secure}.driver`。列出这些文件是为了突出完整性。执行某个驱动程序时，**仅需使用 `secure.driver` 或 `名称-secure.driver` 即可**。该驱动程序会自动调用相关的驱动程序。

Solaris Security Toolkit 体系结构包括可以使 `driver`、`finish` 和 `audit` 脚本不经修改（不修改实际的脚本本身）便可用于不同环境的配置信息。在 `finish` 和 `audit` 脚本中使用的所有变量都在一个配置文件集中进行维护。这些配置文件是由驱动程序导入的，当驱动程序调用 `finish` 和 `audit` 脚本时配置文件使这些脚本可以使用变量。

Solaris Security Toolkit 软件具有四个主要配置文件，它们都存储在 **Drivers** 目录中：

- `driver.init`
- `finish.init`
- `user.init`
- `user.run`

`user.run` 文件为您提供了一个位置，可用于写入 Solaris Security Toolkit 函数的替换版本或加强版本；如果存在，则会自动使用这些版本。



注意 - 只能更改 `user.init` 配置文件中的变量定义，不可更改 `driver.init` 和 `finish.init` 配置文件中的变量定义。

由驱动程序调用的 `finish` 脚本位于 `Finish` 目录中。由驱动程序调用的 `audit` 脚本位于 `Audit` 目录中。由驱动程序安装的 `Files` 从 `Files` 目录中读取。有关 `finish` 脚本的更多信息，请参阅《Solaris Security Toolkit 4.2 Reference Manual》中的第 4 章。有关 `audit` 脚本的更多信息，请参阅《Solaris Security Toolkit 4.2 Reference Manual》中的第 5 章。

图 1-2 是驱动程序控制流程的流程图。

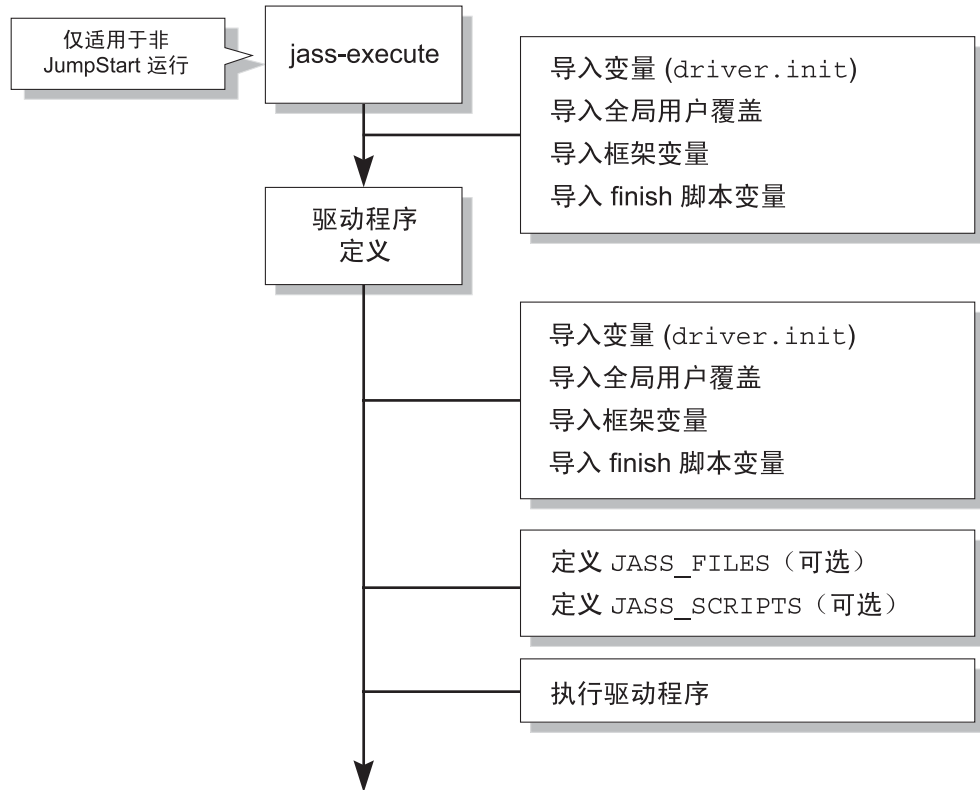


图 1-2 驱动程序控制流程

1. 仅适用于非 **JumpStart** 运行，驱动程序将运行 `jass-execute` 命令。**JumpStart** 运行直接调用驱动程序，而不是调用 `jass-execute` 命令。

2. 驱动程序可以明确地设置变量。
3. 驱动程序从不同的 `.init` 文件中导入所有的环境变量。
4. 驱动程序定义 `JASS_FILES` 和 `JASS_SCRIPTS` 环境变量。这两个定义是可选的；可以定义任意一个单独的环境、也可以定义两个或者不定义任何环境。
有关定义 `JASS_FILES` 和 `JASS_SCRIPTS` 环境变量的更多信息，请参见《**Solaris Security Toolkit 4.2 Reference Manual**》中的第 7 章。
5. 驱动程序调用 `driver.run` 来执行由 `JASS_FILE` 和 `JASS_SCRIPTS` 环境变量所定义的任务。
6. (可选) 驱动程序定义特定的驱动程序行为，可用于覆盖 `finish.init` 或 `user.init` 中的系统默认值。代码示例 1-1 中的驱动程序明确地将 `JASS_PASS_HISTORY` 设置为 4。

代码示例 1-1 说明了驱动程序控制流程代码。

代码示例 1-1 驱动程序控制流程代码

```
DIR="/bin/dirname $0`"
JASS_PASS_HISTORY="4"
export DIR
. ${DIR}/driver.init

JASS_FILES="
                                /etc/cron.d/cron.allow
                                /etc/default/ftpd
                                /etc/default/telnetd
"

JASS_SCRIPTS="
                                install-at-allow.fin
                                remove-unneeded-accounts.fin
"
. ${DIR}/driver.run
```

1. 该代码实例设置 `DIR` 环境变量并将其导出，以使驱动程序识别启动目录。
2. 驱动程序明确地将 `JASS_PASS_HISTORY` 环境变量设置为 4。
3. 驱动程序读取不同的 `.init` 文件（从 `driver.init` 开始）。
4. 将 `JASS_FILES` 环境变量定义为包含从 `JASS_HOME_DIR/Files` 目录复制到客户机的那些文件。
5. 使用由 **Solaris Security Toolkit** 软件运行的 `finish` 脚本来定义 `JASS_SCRIPTS` 环境变量。

6. 通过调用 `driver.run` 驱动程序执行运行加强系统安全性的操作。`driver.run` 将复制 `JASS_FILES` 中指定的文件，并运行 `JASS_SCRIPTS` 中指定的脚本。

Files 目录

`JASS_FILES` 环境变量和 `driver.run` 脚本使用本目录来存储被复制到 **JumpStart** 客户机的文件。

该目录中包含以下文件：

- `/.cshrc`
- `/.profile`
- `/etc/default/sendmail`
- `/etc/dt/config/Xaccess`
- `/ftpd/banner.msg`
- `/etc/hosts.allow`
- `/etc/hosts.allow-15k_sc`
- `/etc/hosts.allow-server`
- `/etc/hosts.allow-suncluster`
- `/etc/hosts.deny`
- `/etc/init.d/klmmod`
- `/etc/init.d/nddconfig`
- `/etc/init.d/set-tmp-permissions`
- `/etc/init.d/sms_arpconfig`
- `/etc/init.d/swapadd`
- `/etc/issue`
- `/etc/motd`
- `/etc/opt/ipf/ipf.conf`
- `/etc/opt/ipf/ipf.conf-15k_sc`
- `/etc/opt/ipf/ipf.conf-server`
- `/etc/security/audit_class+5.10`
- `/etc/security/audit_class+5.8`
- `/etc/security/audit_class+5.9`
- `/etc/security/audit_control`
- `/etc/security/audit_event+5.10`
- `/etc/security/audit_event+5.8`
- `/etc/security/audit_event+5.9`
- `/etc/sms_domain_arp`
- `/etc/sms_sc_arp`
- `/etc/syslog.conf`
- `/root/.cshrc`
- `/root/.profile`
- `/var/opt/SUNWjass/BART/rules`
- `/var/opt/SUNWjass/BART/rules-secure`

Finish 目录

该目录包含 **finish** 脚本，该脚本在执行过程中进行系统修改和更新。该目录中的脚本分为以下类别：

- 禁用
- 启用
- 安装
- 最小化
- 打印
- 删除
- 设定
- 更新

有关每个类别中脚本的详细列表和每个脚本的说明，请参阅《Solaris Security Toolkit 4.2 Reference Manual》。

OS 目录

该目录仅包含 Solaris OS 映像。在 JumpStart 软件安装进程进行客户机安装时，将这些映像作为 Solaris OS 源使用。如果目录名称遵循了 Solaris Security Toolkit OS 命名约定，则 `add_client` 脚本将接受包含在该目录中的 Solaris OS 版本作为参数。

有关加载和修改 Solaris OS 映像的详细信息，请参阅 Sun BluePrints 书籍《JumpStart Technology: Effective Use in the Solaris Operating Environment》。

标准安装命名约定如下。

Solaris OS

对于 Solaris OS，请使用以下的命名标准：

`Solaris_OS 版本_CD 版本的 4 位数年份-2 位数月份`

例如，日期为 2005 年 3 月的 Solaris 10 Operating Environment CD 的目录名称应为 `Solaris_10_2005-03`。通过将 Solaris OS 的更新和版本分开，可以对测试和开发目的维护进行非常精细的控制。

适用于 x86/x64 平台的 *Solaris OS*

对于适用于 x86/x64 平台的 Solaris OS，请使用以下目录命名标准：

`Solaris_OS 版本_CD 版本的 4 位数年份-2 位数月份_ia`

例如，如果适用于 x86/x64 平台的 Solaris OS 发布日期为 2005 年 3 月，则该目录名称为：`Solaris_10_2005-03_ia`。

Packages 目录

该目录包含可以使用 **finish** 脚本进行安装并通过 **audit** 脚本进行验证的软件包。例如，**Open Secure Shell** 软件包可以存储在 **Packages** 目录中，这样适当的 **finish** 脚本会根据需要安装该软件。

Solaris Security Toolkit 软件包含的几个 **finish** 和 **audit** 脚本可以执行软件安装、基本的配置和验证功能。从 **Packages** 目录中安装和验证软件的脚本包括：

- `install-fix-modes.{fin|aud}`
- `install-jass.{fin|aud}`
- `install-md5.{fin|aud}`
- `install-openssh.{fin|aud}`

Patches 目录

本目录用于存储 **Solaris OS** 的 **Recommended and Security Patch Clusters**。下载必需的修补程序并将其提取到本目录中。

通过将修补程序放置在该目录中并在其中提取，可以简化安装操作。当修补程序提取到该目录中时，**Solaris Security Toolkit** 软件修补程序安装脚本会使安装自动化，因此无需手动为每个系统安装提取修补程序群集。

为所使用的每个 **Solaris OS** 版本创建子目录。例如，**Patches** 目录中可能已经存在 `9_Recommended` 和 `10_Recommended` 目录。

Solaris Security Toolkit 软件支持适用于 **x86/x64** 平台的 **Solaris OS** 修补程序群集。这些修补程序群集所支持的命名约定与通过 **SunSolve OnLineSM** 服务获得的命名约定相同。

格式为 `<release>_x86_Recommended`。针对 **Solaris 10 OS** 的适用于 **x86/x64** 平台的 **Solaris OS** 修补程序群集会位于一个名为 `10_x86_Recommended` 的目录中。

Profiles 目录

该目录包含所有 **JumpStart** 配置文件。**JumpStart** 软件使用这些配置文件中包含的配置信息来确定安装的 **Solaris OS** 群集（例如，核心、最终用户、开发人员或完整分发）、磁盘布局以及要执行的安装类型（例如，独立安装）。

`rules` 文件列出了它使用的 **JumpStart** 配置文件，以定义如何构建特定的系统或系统组。

Sysidcfg 目录

与 Profiles 目录类似，Sysidcfg 目录仅包含在 JumpStart 模式安装过程中使用的文件。通过提供必需的安装信息，这些文件可使 Solaris OS 安装自动化。特定操作系统的信息存储在单独的目录树中。

每个 Solaris OS 都具有一个单独的目录。每个版本都有一个名为 Solaris_OS 版本的目录。Solaris Security Toolkit 软件提供了用于 Solaris OS 版本 2.5.1 至 10 的范例 sysidcfg 文件。

可将范例 sysidcfg 文件扩展为其他类型，例如，每个网络或主机。Solaris Security Toolkit 软件支持任意 sysidcfg 文件。

有关 sysidcfg 文件的其他信息，请参见 Sun BluePrints 书籍《JumpStart Technology: Effective Use in the Solaris Operating Environment》。

数据信息库

尽管不处于 JASS_HOME_DIR 目录结构中，数据系统信息库（或者 JASS_REPOSITORY 目录）支持 Solaris Security Toolkit 撤消运行操作，根据每个运行操作的执行方式来保存数据，维护由软件修改的文件清单以及保存执行日志的数据。该目录位于 /var/opt/SUNWjass/runs/时间戳中。

维护版本控制

维护由 Solaris Security Toolkit 软件使用的所有文件和脚本的版本控制非常关键，有以下两个原因：

1. 该环境的目标之一就是能够重新创建系统安装。如果没有安装过程中所用到所有文件版本的快照，那么该目标是不可能实现的。
2. 由于这些脚本执行安全性功能（这对很多组织来说都是关键的过程），因此必须格外小心以确保仅实施必要的且经过测试的更改。

在 Solaris OS SUNWspot 软件包中提供了源代码控制系统 (SCCS) 的版本控制软件包。用户可以使用来自免费软件和商业供应商的其他版本控制软件来管理版本信息。无论您使用哪种版本控制产品，都要使用一个过程，从而为将来系统的重新创建管理更新信息并获取版本信息。

除版本控制外，还要使用完整性管理解决方案以确定文件的内容是否已经被修改。尽管系统的特权用户也许能够绕过版本控制系统，但他们无法轻易地绕过完整性管理系统，该系统在远程系统上维护其完整性数据库。进行集中管理时，完整性管理解决方案会充分发挥其作用，因为本地存储的数据库可能会被恶意修改。

配置和定制 Solaris Security Toolkit 软件

Solaris Security Toolkit 软件包含用于脚本、框架功能和变量的默认值，这些默认值可实现 Sun BluePrints 书籍《Enterprise Security: Solaris Operating Environment Security Journal, Solaris Operating Environment Versions 2.5.1, 2.6, 7, and 8》以及关于安全性的 Sun BluePrints OnLine 文章中的所有安全准则。这些设置并不适用于所有系统，因此必须定制 Solaris Security Toolkit 软件来满足系统的安全性要求。

Solaris Security Toolkit 软件的一个最重要特征就是可以轻松地定制，使它符合您的环境、系统 and 安全性要求。要定制 Solaris Security Toolkit 软件，请通过驱动程序、finish 脚本、audit 脚本、框架功能、环境变量以及文件模板来对其进行调整。

大多数用户不需要修改 Solaris Security Toolkit 代码。如果在您的环境中使用 Solaris Security Toolkit 软件必须修改代码，请将该代码复制到 user.run 中的一个唯一的函数名称中，以便轻松地跟踪更改，如第 12 页“准则”中所述。

在整个指南中，提供了用于定制 Solaris Security Toolkit 软件的准则和指导。请参阅《Solaris Security Toolkit 4.2 Reference Manual》，以查找关于定制驱动程序的帮助信息。定制包括修改和创建文件或变量。

本指南还提供了用于定制 Solaris Security Toolkit 软件的实例。实例重点介绍了定制 Solaris Security Toolkit 软件的某些方法；然而，还存在许多其他方法。

在尝试定制 Solaris Security Toolkit 软件之前，必须准确理解以下各节所给出的信息。这些信息基于从很多部署方案收集的共享经验，因此可以避免通常所犯的错误。

策略和要求

定制和部署 Solaris Security Toolkit 软件时，适当的规划可以确保正确配置结果平台配置，并且与组织的期望相一致。

在规划阶段中，请务必从各种来源中获取输入，这些来源包括安全性策略和标准、工业规程和准则以及供应商提供的首选做法。

除此之外非常重要的一点是，应该考虑应用程序和操作要求以确保配置结果不会影响平台为其预期的业务功能提供服务。

准则

定制 Solaris Security Toolkit 软件时，请考虑以下准则。了解并遵守这些准则有助于使维持部署的过程更加简单、有效。

- 作为一般规则，请勿更改由 **Solaris Security Toolkit** 软件提供的任何原始文件（驱动程序、脚本、文件等等）。更改原始文件将阻止并限制您的组织升级到 **Solaris Security Toolkit** 软件最新版本的能力，因为对原始文件的任何更改都可能被该文件的新版本所覆盖。（定制的所有更改都将丢失，而系统配置的更改方式可能不合适。）要定制其中任何文件，请首先制作一个副本，然后修改该副本，保留原始文件不变。对于该规则只存在一个例外：
 - `sysidcfg` 文件
- **Solaris Security Toolkit 4.2** 软件新增了一个特性，使您可以对 `Files` 目录中的模板使用关键字后缀。这样，系统管理员就不必修改 **Solaris Security Toolkit 4.2** 软件中的任何默认模板，而只在需要时使用后缀即可。
- 对驱动程序或脚本的副本进行命名，以便区别于其原始版本。使用可表明脚本用途的前缀或关键字。例如，包含公司的名称或股票标记、部门标识符或平台、应用程序类型的前缀都是不错的命名标准。表 1-1 列出了几个命名标准的实例。

表 1-1 定制文件的命名标准

定制文件	命名标准
<code>abccorp-secure.driver</code>	公司前缀
<code>abcc-nj-secure.driver</code>	公司股票代码、位置
<code>abccorp-nj-webserver.driver</code>	公司、位置、应用程序类型
<code>abc-nj-trading-webserver.driver</code>	公司、位置、组织、应用程序类型

- 查看以下 **Solaris Security Toolkit** 文件是否适合您的系统。要定制这些文件，请复制原始文件、将副本重命名为 `user.init` 和 `user.run`，然后对副本的内容进行修改或添加。

<code>Drivers/user.init.SAMPLE</code>	用于定制全局参数
<code>Drivers/user.run.SAMPLE</code>	用于定制全局函数

注 - 请注意，如果您使用 `pkgrm` 命令删除了 `SUNWjass`，由其创建的 `user.init` 和 `user.run` 文件不会被删除。这种情况同样也会发生在其他不包含在原始软件分发版中的、后来添加到 **Solaris Security Toolkit** 目录结构中的用户文件。

注 - **Solaris Security Toolkit 4.2** 软件中对 `pkgrm` 命令做了新的改进。在这一版本中，`pkgrm` 命令的第一步是检查分发中所有文件的完整性。如果存在任何不同的文件，则 `pkgrm` 命令会退出，并显示一个错误消息，提示系统管理员放置正确的文件或删除已修改的文件。

第2章

保护系统安全：方法论

本章提供保护系统安全的方法论。在使用 **Solaris Security Toolkit** 软件保护系统安全之前，您可以应用该软件的过程。

本章包含以下主题：

- 第 15 页 “规划与准备”
- 第 26 页 “开发和执行 **Solaris Security Toolkit** 配置文件”
- 第 26 页 “安装软件”
- 第 28 页 “验证应用程序和服务功能”
- 第 29 页 “维护系统安全”

规划与准备

适当的规划是成功使用 **Solaris Security Toolkit** 软件来保护系统安全的关键。规划阶段将根据组织的安全策略和标准以及系统的应用程序和操作要求，为系统构建一个 **Solaris Security Toolkit** 配置文件。该阶段分为以下任务：

- 第 15 页 “考虑风险和收益”
- 第 17 页 “查看安全策略、标准和相关文档”
- 第 18 页 “确定应用程序和服务要求”

尽管未在本书中说明，该阶段的其他注意事项可能包括了解风险和收益，了解基础结构及其安全要求，考虑责任、日志以及对使用的审计。

考虑风险和收益

加强系统的安全性时，必须采取特殊的预防措施以确保在使用 **Solaris Security Toolkit** 软件之后，系统可以正常运行。此外，优化该过程来确保尽可能短的停机时间也非常重要。

注 – 在保护经过部署的系统的的天时，在某些情况下，重新构建系统可能是更加有效的方法，这样可在重新安装时加强系统的安全性，然后重新加载操作所需的全部软件。

本节给出了在尝试保护系统安全之前必须准确理解的注意事项。请仔细权衡风险与收益以确定适合您的组织的操作。

1. 了解系统上的服务和应用程序的要求。

在运行 **Solaris Security Toolkit** 软件之前，必须识别运行在系统上的服务和应用程序。必须列举出所有与服务 and 应用程序相关的依赖性，以便 **Solaris Security Toolkit** 软件可以有效地进行调整。未执行此操作可能会禁用服务或阻止必需的服务启动。因为由 **Solaris Security Toolkit** 软件所做的更改在大多数情况下都可以撤消，在安装之前开发正确的配置文件可以减少在实现 **Solaris Security Toolkit** 软件时可能造成的停机时间。

2. 要考虑到系统必须脱机和重新引导的情况。

要使由 **Solaris Security Toolkit** 软件所做的更改生效，必须重新引导系统。根据系统的重要程度、系统所提供的服务、维护窗口的可用性，组织可能在执行该软件时遇到不同的困难。在仔细权衡停机时间的花费与不增强安全性的风险之后，必须做出一个决定。

3. 系统可能要求多次重新引导以验证其功能性。

只要有可能，就应在将更改应用到影响关键任务的实际环境之前，首先在非生产系统中进行试验。但有时无法满足这样的要求，例如，没有足够的硬件和软件对目标环境进行有效的镜像。在对 **Solaris Security Toolkit** 软件执行加强安全性操作之前和之后都必须进行测试。在系统加强安全性后，可能仍然存在未识别的、需要进行错误诊断的依赖性。在大多数情况下，可以使用本章中介绍的技术非常快速地解决这些问题。如果在运行 **Solaris Security Toolkit** 软件后发现了功能问题，可能需要额外的平台重新引导来撤消 **Solaris Security Toolkit** 软件的影响或者对系统的配置做进一步的更改以支持并启用缺失的功能。

4. 平台安全性需要的不仅仅是加强和审计。

当考虑改进系统配置以增强其安全性时，必须了解的是，平台加强和审计只代表了可以用于和应该用于保护系统、服务和数据的一小部分措施。其他措施和控制不在本文的范围之内，但希望您考虑到与帐户管理、权限管理、文件系统和数据完整性、基于主机的访问控制、入侵检测、安全漏洞扫描和分析以及应用程序安全相关的问题。

5. 系统中可能存在可被利用的安全漏洞或者已经被利用。

正在加强安全性的平台可能已经被攻击者利用。**Solaris Security Toolkit** 软件的执行可能已经太晚，因而无法为已被利用的安全漏洞提供保护。在存在已被利用的安全漏洞的情况下，请：

a. 重新安装系统。

b. 安装 **Solaris Security Toolkit** 软件。

- c. 使用 Solaris Security Toolkit 软件来增强系统安全性。

查看安全策略、标准和相关文档

保护系统安全的首要任务就是了解组织的相关安全策略、标准以及关于平台安全性的准则。应将这些文档作为 **Solaris Security Toolkit** 配置文件的基础，因为这些文档提供了对于组织内的所有系统要遵循的要求和做法。如果您的组织没有这些文档，则开发它们可以提高定制 **Solaris Security Toolkit** 软件的能力。

注 – 当查找这些文档时，请记住某些材料可能列于最佳做法或其他文档中。

有关安全策略的更多信息，请参阅 **Sun BluePrints OnLine** 文章 "**Developing a Security Policy**"。该文档可以使您对安全策略在组织安全性计划中发挥的作用有更进一步的了解。

以下两个实例说明了策略声明是如何直接影响 **Solaris Security Toolkit** 配置文件的配置方式的。

实例 1

- **策略** – 一个组织必须使用支持强大的用户身份验证和传输数据加密的管理协议。
- **配置文件影响** – 不应该使用明文协议，例如：**Telnet**、文件传输协议 (**FTP**)、简单网络管理协议版本 1 (**SNMPv1**) 和其他明文协议。**Solaris Security Toolkit** 中的 `secure.driver` 会禁用这些服务，因此不需要额外的配置。

注 – 可以使用扩展（例如 **Kerberos**）将 **Telnet** 和 **FTP** 服务配置为支持更强的身份验证和加密。但是，它们的默认配置不支持这些添加的安全级别。

实例 2

策略 – 强制所有用户每 30 天更改一次密码。

配置文件影响 – 可以将 **Solaris Security Toolkit** 软件配置为启用密码过期。**Solaris Security Toolkit** 软件中的 `secure.driver` 将密码的最长过期时间设置为 8 周（56 天）。要符合该策略，必须更改 **Solaris Security Toolkit** 软件的配置文件。请参阅《**Solaris Security Toolkit 4.2 Reference Manual**》。

尽管在系统中运行时，**Solaris Security Toolkit** 软件中的 **secure.driver** 会启用密码过期，但是在现有用户更改密码之前，这一更改不会对他们产生影响。要为现有用户启用密码过期，请在每个用户帐户上调用 **passwd(1)** 命令。要强制现有用户更改密码，您可以使用 **passwd -f** 命令。有关 **passwd(1)** 命令的更多信息，请参阅《**Solaris 10 OS Reference Collection**》。

确定应用程序和服务要求

该任务确保在系统加强安全性后服务仍然能够保持正常工作。该任务包括以下步骤：

- 第 18 页 “确定应用程序和可用服务清单”
- 第 18 页 “确定服务要求”

确定应用程序和可用服务清单

清点应用程序、服务和可用功能或管理功能。该清单对于确定实际在系统上使用的软件是必需的。在很多情况下，系统会配置比使用的软件更多的软件，甚至配置有不支持业务功能的软件。

应该构建尽可能小的系统。也就是说，**不应该**安装与支持业务功能无关的软件。系统上不必要的软件应用程序会为攻击者利用系统提供更多的机会。此外，系统上安装更多的软件通常就等于必须要应用更多的修补程序。有关最小化 **Solaris OS** 的信息，请参阅 **Sun BluePrints OnLine** 文章 "**Minimizing the Solaris Operating Environment for Security**"。有关最小化 **Sun Fire** 系统域的信息，请参阅 **Sun BluePrints Online** 文章 "**Part I: Minimizing Domains for Sun Fire V1280, 6800, 12K, and 15K Systems**" 和 "**Part II: Minimizing Domains for Sun Fire V1280, 6800, 12K, and 15K Systems**"。

当构建软件的清单时，请确保除包括位于系统上的应用程序之外，还要包括基础结构组件（例如管理、监视以及备份软件）。

确定服务要求

在完成应用程序和服务清单后，应确定加强安全性的过程是否会影响到组件的依赖性。很多第三方应用程序不直接使用由 **Solaris OS** 提供的服务。对于直接使用 **Solaris OS** 提供的服务的应用程序，以下各节提供了有用的信息。

- 第 19 页 “共享库”
- 第 21 页 “配置文件”
- 第 22 页 “服务框架”

注 – 本节中的所有实例均来自 **Solaris 9 OS**。

共享库

了解支持应用程序需要的库是非常重要的。在调试环境时这种知识最为有用，在准备要进行安全性加强的系统时也很有用。当系统状态未知时，应收集尽可能多的信息以便准确地理解问题（例如，软件相关性）。

您可以使用三种方法来确定应用程序所使用的库，这取决于所安装的 **Solaris OS** 版本。本节为每种方法提供了一个代码实例。

- **方法 1** — 获取文件系统对象（如应用程序二进制文件或库）的信息（代码示例 2-1）。
- **方法 2** — 收集正在运行的进程的信息以分析正在运行的应用程序（代码示例 2-2）。
- **方法 3** — 标识动态加载的应用程序，以便在程序启动时对其进行跟踪（代码示例 2-3）。

方法 1

要获取有关文件系统对象的信息，请使用 `/usr/bin/ldd` 命令。

例如：确定支持域名系统 (DNS) 服务器软件所需要的库。

代码示例 2-1 获得有关文件系统对象的信息

```
# ldd /usr/sbin/in.named
libresolv.so.2 => /usr/lib/libresolv.so.2
libsocket.so.1 => /usr/lib/libsocket.so.1
libnsl.so.1 => /usr/lib/libnsl.so.1
libc.so.1 => /usr/lib/libc.so.1
libdl.so.1 => /usr/lib/libdl.so.1
libmp.so.2 => /usr/lib/libmp.so.2
/usr/platform/SUNW,Ultra-5_10/lib/libc_psr.so.1
```

方法 2

要从正在运行的进程中收集信息，请使用 `/usr/proc/bin/pldd` 命令（在 **Solaris OS** 版本 8、9 和 10 中可用）。

代码示例 2-2 从正在运行的进程中收集信息

```
# pldd 20307
20307: /usr/sbin/in.named
/usr/lib/libresolv.so.2
/usr/lib/libsocket.so.1
/usr/lib/libnsl.so.1
/usr/lib/libc.so.1
/usr/lib/libdl.so.1
/usr/lib/libmp.so.2
/usr/platform/sun4u/lib/libc_psr.so.1
/usr/lib/dns/dnssafe.so.1
/usr/lib/dns/cylink.so.1
```

方法 3

除应用程序所链接的库之外，`pldd` 命令还显示由应用程序动态加载的共享库。也可以使用以下 `truss` 命令来收集该信息。

注 - 为简洁起见，本例对输出内容做了删节。

代码示例 2-3 确定动态加载的应用程序

```
# truss -f -topen,open64 /usr/sbin/in.named
20357: open("/usr/lib/libresolv.so.2", O_RDONLY) = 3
20357: open("/usr/lib/libsocket.so.1", O_RDONLY) = 3
20357: open("/usr/lib/libnsl.so.1", O_RDONLY) = 3
20357: open("/usr/lib/libc.so.1", O_RDONLY) = 3
20357: open("/usr/lib/libdl.so.1", O_RDONLY) = 3
20357: open("/usr/lib/libmp.so.2", O_RDONLY) = 3
20357: open("/usr/lib/nss_files.so.1", O_RDONLY) = 4
20357: open("/usr/lib/nss_files.so.1", O_RDONLY) = 4
20357: open("/usr/lib/dns/dnssafe.so.1", O_RDONLY) = 4
20357: open("/usr/lib/dns/cylink.so.1", O_RDONLY) = 4
20357: open("/usr/lib/dns/sparcv9/cylink.so.1", O_RDONLY) = 4
```

此输出的版本包含进程标识符、系统调用（在本例中为 `open`）及其变量以及系统调用的返回值。使用返回值，可清晰地看到系统调用成功找到和打开了共享库。

了解共享库列表后，请使用以下命令来确定它们属于哪个 **Solaris OS** 软件包。

```
# grep "/usr/lib/dns/cylink.so.1" /var/sadm/install/contents
/usr/lib/dns/cylink.so.1 f none 0755 root bin 63532 24346 \
1018126408 SUNWcs1
```

结果输出表示该共享库属于 `SUNWcs1`（核心，共享库）软件包。当执行平台最小化时该进程尤其有用，因为它有助于确定支持应用程序或服务所必需的软件包。

配置文件

另一种收集要求的方法就是通过配置文件进行。由于可以重命名或删除配置文件以禁用服务，因此该过程更直接地影响系统的安全性加强方式。有关更多信息，请参阅《**Solaris Security Toolkit 4.2 Reference Manual**》。

要确定是否正在使用配置文件，请使用 `truss` 命令。

注 – 为简洁起见，本例对输出内容做了删节。

代码示例 2-4 确定是否正在使用配置文件

```
# truss -f -topen,open64 /usr/sbin/in.named 2>&1 | \
grep -v "/usr/lib/*.so.*"
20384: open("/etc/resolv.conf", O_RDONLY) = 3
20384: open("/dev/conslog", O_WRONLY) = 3
20384: open("/usr/share/lib/zoneinfo/US/Eastern", O_RDONLY) = 4
20384: open("/var/run/syslog_door", O_RDONLY) = 4
20384: open("/etc/nsswitch.conf", O_RDONLY) = 4
20384: open("/etc/services", O_RDONLY) = 4
20384: open("/etc/protocols", O_RDONLY) = 4
20384: open("/etc/named.conf", O_RDONLY) = 4
20384: open("named.ca", O_RDONLY) = 5
20384: open("named.local", O_RDONLY) = 5
20384: open("db.192.168.1", O_RDONLY) = 5
20384: open("db.internal.net", O_RDONLY) = 5
```

在本例中，DNS 服务使用了配置文件，例如 `/etc/named.conf`。正如前面的实例中介绍的，如果服务的返回值表明一个错误，则可能存在问题。仔细记录加强安全性前后的结果有助于加速整个验证过程。

服务框架

该类别包括框架或元服务，基于它们可以构建更大、更复杂的应用程序。通常可在该类别中找到的框架类型有：

- 命名服务，例如：网络信息服务 (NIS)、NIS+ 和轻量目录访问协议 (LDAP)
- 验证服务，例如：Kerberos 和 LDAP
- 实用程序服务，例如：远程过程调用 (RPC) 实用程序使用的端口映射器

通常很难始终清楚地了解应用程序何时依赖于这些类型的服务。当需要特殊的调整来配置应用程序时（例如将其添加到 Kerberos 区域），相关性是已知的。但是，应用程序依赖性并非总是需要添加任务，而且供应商可能不会记录实际的依赖性。

RPC 端口映射器就是一个这样的实例。Solaris Security Toolkit 软件中的 `secure.driver` 会禁用 RPC 端口映射器。该操作可能会引起依赖于该服务的其他服务中的异常行为。基于过去的经验，服务退出、挂起或者失败取决于应用程序中处理异常情况的代码是否编写得完善。要确定应用程序是否正在使用 RPC 端口映射器，请使用 `rpcinfo` 命令。例如：

代码示例 2-5 确定哪些应用程序正在使用 RPC

```
# rpcinfo -p
100000    3    tcp    111    rpcbind
100000    4    udp    111    rpcbind
100000    2    udp    111    rpcbind
100024    1    udp    32777  status
100024    1    tcp    32772  status
100133    1    udp    32777
100133    1    tcp    32772
100021    1    udp    4045   nlockmgr
100021    2    udp    4045   nlockmgr
100021    3    udp    4045   nlockmgr
100021    4    udp    4045   nlockmgr
100021    1    tcp    4045   nlockmgr
```

服务列中引用了来自 `/etc/rpc` 文件或已配置的命名服务（例如 LDAP）的信息。

如果该文件没有用于某项服务（通常是第三方产品）的条目，则该服务字段可能为空。这就使确定由其他应用程序注册的应用程序变得更加困难。

例如，考虑 `rusers` 命令。该命令依赖于 **RPC** 端口映射服务。如果 **RPC** 端口映射器没有运行，则 `rusers` 命令显示为挂起。程序最终会超时，并显示以下错误消息：

```
# rusers -a localhost
localhost: RPC: Rpcbnd failure
```

出现此问题是由于该程序无法与服务进行通信。在从 `/etc/init.d/rpc` 启动 **RPC** 端口映射服务之后，该程序可立即产生其结果。

作为另一个实例，请考虑 **RPC** 端口映射服务正在运行，而没有将 `rusers` 服务配置为运行的情况。在这种情况下，就会产生完全不同的响应，并且可以相对简单地进行验证。

代码示例 2-6 验证 `rusers` 服务

```
# rusers -a localhost
localhost: RPC: Program not registered
# grep rusers /etc/rpc
rusersd          100002  rusers
# rpcinfo -p | grep rusers
<No output generated>
```

假设 `rpcinfo` 命令没有用于 `rusers` 服务的注册表，可以肯定地认为没有将该服务配置为运行。这种假设可以通过查看 `/etc/inet/inetd.conf` 中的服务条目进行验证。

```
# grep rusers /etc/inet/inetd.conf
# rusersd/2-3  tli      rpc/datagram_v,circuit_v  wait root
/usr/lib/netsvc/rusers/rpc.rusersd  rpc.rusersd
```

该服务行开头的注释标记 (#) 表明已禁用 `rusers` 服务。要启用该服务，请取消注释行并将 `SIGHUP` 信号发送到 `/usr/sbin/inetd` 进程，如下所示。

```
# pkill -HUP inetd
```

注 — 仅在 Solaris OS 版本 7 至 10 中提供了 `pkill` 命令。对于其他版本，请分别使用 `ps` 和 `kill` 命令查找和使用信号通知该进程。

另一种确定应用程序是否使用 **RPC** 工具的方法是使用前面介绍的 `ldd` 命令。

代码示例 2-7 确定使用 **RPC** 的应用程序的备选方法

```
# ldd /usr/lib/netsvc/rusers/rpc.rusersd
libnsl.so.1 => /usr/lib/libnsl.so.1
librpcsvc.so.1 => /usr/lib/librpcsvc.so.1
libc.so.1 => /usr/lib/libc.so.1
libdl.so.1 => /usr/lib/libdl.so.1
libmp.so.2 => /usr/lib/libmp.so.2
/usr/platform/SUNW,Ultra-250/lib/libc_psr.so.1
```

`librpcsvc.so.1` 条目与文件名指明该服务依赖于 **RPC** 端口映射服务。

除 **RPC** 端口映射器外，应用程序还可能依赖于其他常用的 **OS** 服务，例如 **FTP**、**SNMP** 或网络文件系统 (**NFS**)。可以使用类似的技术来调试这些服务以及确定它们是否确实是支持业务功能所必需的。其中一种方法涉及使用 `netstat` 命令，如下所示。

```
# netstat -a | egrep "ESTABLISHED|TIME_WAIT"
```

该命令返回一个最近使用的服务列表，例如：

表 2-1 列出最近使用的服务

localhost.32827	localhost.32828	49152	0	49152	0
ESTABLISHED					
localhost.35044	localhost.32784	49152	0	49152	0
ESTABLISHED					
localhost.32784	localhost.35044	49152	0	49152	0
ESTABLISHED					
localhost.35047	localhost.35046	49152	0	49152	0
ESTABLISHED					
localhost.35046	localhost.35047	49152	0	49152	0
ESTABLISHED					
filefly.ssh	192.168.0.3.2969	17615	1	50320	0
ESTABLISHED					

在此实例中，正在使用多种服务，但是不清楚哪些服务或应用程序拥有哪些端口。可以通过使用 `pfiles(1)` 命令（在 Solaris OS 版本 8、9 和 10 中可用）对进程进行检查以收集该信息。 `pfiles` 命令报告每个进程中所有打开的文件的信息。

代码示例 2-8 确定哪些服务或应用程序拥有哪些端口

```
# for pid in `ps -a eo pid | grep -v PID`; do
> pfiles ${pid} | egrep "^\${pid}:|sockname:"
> done
```

确定这些依赖性的一种更为有效的方法是使用 `lsof`（列出打开的文件）命令。

从以下网址下载 `lsof` 源代码：

<ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof/>

从以下网址下载 `lsof` 二进制文件：

<http://www.sunfreeware.com>

`lsof` 命令能够确定哪些进程正在使用哪些文件和端口。例如，要确定前面实例中的哪些进程正在使用端口 **35047**，请使用以下命令。

代码示例 2-9 确定哪些进程正在使用文件和端口

```
# ./lsof -i | grep 35047
ttsession    600 root 9u  IPv4 0x3000b4d47e8      0t1  TCP
localhost:35047->localhost:35046 (ESTABLISHED)
dtexec       5614 root 9u  IPv4 0x3000b4d59e8      0t0  TCP
localhost:35046->localhost:35047 (ESTABLISHED)
```

`lsof` 的输出表明端口 35047 正在用于 `dtexec` 和 `ttsession` 进程之间的通信。

使用 `lsof` 程序，您可以更迅速地确定系统间或应用程序间的需要使用文件系统或网络的依赖性。在本节中阐述的几乎所有内容都可以使用 `lsof` 程序来获取。

注 – 此处介绍的确定依赖性的方法可能无法找出那些极少使用的依赖性。除使用这些方法外，还请查看 **Sun** 文档和供应商提供的文档。

开发和执行 Solaris Security Toolkit 配置文件

在完成规划和准备阶段后，即可开发和执行安全性配置文件。安全性配置文件包括相关配置、加强安全性和安全性驱动程序，例如，

名称-`{config|hardening|secure}.driver`、脚本以及实现站点特定的安全策略的文件。

您可以对 **Solaris Security Toolkit** 软件附带的一个安全性配置文件进行配置，也可以自行开发。每个组织的策略、标准和应用程序都不会相同，即使只有细微的差别。

要定制一个安全性配置文件，请通过 **finish** 脚本、**audit** 脚本、环境变量、框架功能以及文件模板来调整该文件的操作。

有关更多信息，请参见以下各章节：

- 有关定制软件的重要准则，请参见第 1 章，第 12 页“配置和定制 **Solaris Security Toolkit** 软件”。
- 有关创建安全性配置文件的实例情况，请参见第 7 章，第 95 页“创建安全性配置文件”。
- 有关定制驱动程序的信息，请参见《**Solaris Security Toolkit 4.2 Reference Manual**》。

如有需要，可参见《**Solaris Security Toolkit 4.2 Reference Manual**》的其他章节以获得有关脚本、框架功能、环境变量和文件的信息。您可能要定制的两个关键环境变量是 `JASS_FILES` 和 `JASS_SCRIPTS`。

要跨越大多数平台强制执行标准，同时保证特定平台之间的差异，请使用嵌套或分层安全性配置文件技术。有关更多信息，请参阅《**Solaris Security Toolkit 4.2 Reference Manual**》。比较结果配置文件与您组织的策略、标准和要求以确保不存在粗心大意或错误的更改。

安装软件

Solaris Security Toolkit 软件的安装对于经过部署的或正在安装的新系统来说是相同的。有关详细指导，请参阅第 3 章。

对于经过部署的系统，一些特定的例子可以使该过程更简单、迅速。这些例子的重点不在于加强安全性过程，而是在于安装前后的任务。

执行安装前任务

在加强经过部署的系统的的天性之前，请考虑和计划两项重要的任务：

- 备份
- 验证

这两个任务有助于确定已部署系统的状态，并且有助于在加强系统安全性之前解决所有潜在的配置问题。

备份数据

该任务重点在于意外事故的规划。发生问题时，必须确保系统的配置和数据以某种形式进行了归档。您**必须**：

- 备份系统
- 确保可以读取备份介质
- 验证是否可以恢复其内容

在对系统配置进行任何重要更改之前，均应采用这些步骤。

验证系统稳定性

验证任务几乎与备份任务同等重要。在实现任何配置更改（例如，由加强安全性过程所做的更改）之前，验证过程可确保系统处于稳定工作状态。此验证过程包括：

- 重新引导
- 成功地测试所有应用程序或服务

尽管已经选择了定义好的测试和验收计划，但计划并不总是可用。如果情况确实如此，请根据系统的使用方式以合理的方式测试该系统。其目的在于确保实际运行的配置与所保存的配置相匹配。

当系统引导或应用程序启动时，查看所显示的所有错误消息或警告。如果**无法**纠正错误，请记录它们，以便在加强安全性过程中不会将其当作问题的潜在原因。当查看日志文件时，请确保包括系统、服务以及应用程序日志，例如：

- `/var/adm/messages`
- `/var/adm/sulog`
- `/var/log/syslog`
- `/var/cron/log`

重新启动系统时如果没有遇到错误或警告消息，或者没有遇到任何**未知**的错误或警告（所有**已知**的都已记录），则该任务完成。系统应该重新启动到一个已知并稳定的状态。如果在验证过程中发现系统正在运行的配置和存储的配置有所不同，请重新评估您组织的更改控制策略和过程以确定导致该情况的原因。

执行安装后的任务

安装后的任务是安装前的任务的延伸。目的在于确保加强系统安全性过程没有对系统或应用程序造成任何新的问题。该任务主要是通过查看系统和应用程序日志文件来完成的。在加强安全性以及随后的重新引导之后创建的日志文件应该与那些在对系统进行加强安全性之前收集的日志文件相似。在某些情况下，由于只启动了较少的服务，因此消息可能会少一些。最重要的一点是，不应该出现新的错误或警告消息。

除查看日志文件外，还要测试功能，因为某些应用程序可能会发生故障但不生成日志条目。有关详细的验证信息，请参见下节。

验证应用程序和服务功能

保护系统安全过程的最后任务会涉及到验证由系统提供的应用程序和服务是否能够正确地运行。该任务还验证安全性配置文件是否成功地满足了安全策略的要求。请在加强安全性的平台启动之后彻底地执行该任务，以确保可检测到任何异常或问题并立即得到纠正。该任务分为两个子任务：验证安全性配置文件安装和验证应用程序和服务功能。

验证安全性配置文件的安装

要验证 **Solaris Security Toolkit** 软件是否正确无误地安装了安全性配置文件，请查看安装日志文件 `jass-install-log.txt`。该文件安装在 `/var/opt/SUWWjass/runs` 中，其所处的目录对于每个加强系统安全性或审计系统安全性的运行（运行的起始时间）都是唯一的。

注 – 请参阅该日志文件以了解 **Solaris Security Toolkit** 软件对系统所进行的操作。对于在系统上运行的每个操作，都会根据运行操作起始时间在目录中保存一个新的日志文件。

除了要验证所安装的配置文件外，还要评估系统的安全配置。执行手动检查或使用工具使该过程自动化。

验证应用程序和服务功能

要对过程应用程序和服务进行验证，请执行已定义好的测试和验收计划。该计划将运行系统或应用程序的各个组件以确定它们是否可用和处于正常工作状态。如果这样的规划不可用，则要根据系统的运行方式以合理的方法对系统进行测试。其目的在于确保安全性加强安全性过程绝对不会影响应用程序或服务执行其功能的能力。

如果发现在加强系统安全性后，应用程序或服务出现故障，请通过查看应用程序日志文件来确定该问题。在很多情况下，可以使用 `truss` 命令来确定应用程序遇到的问题。在知道问题所在后，可以针对该问题并跟踪返回到 **Solaris Security Toolkit** 软件所做的更改。

维护系统安全

很多组织通常会犯的一个错误就是仅在安装过程中强调安全，然后就很少或不再重新考虑它。维护安全是一个实时进行的过程。必须定期查看和重新考虑系统安全。

维护一个安全的系统要求提高警惕，因为任何系统的安全配置随着时间的推移都会逐渐地公开。例如，系统安全漏洞会变得众所周知。

以下的基本指导提供了维护系统安全性的概述：

- 在安装任何修补程序前后都要查看系统的安全状态。同时，保证系统已使用最新的修补程序进行更新也非常重要。

Solaris OS 修补程序可能在安装过程中安装其他软件包，并且会覆盖系统配置文件。**Solaris Security Toolkit** 软件可以协助您应用修补程序，因为它支持在系统上的重复运行操作，所以可以在安装修补程序后保护系统的安全。在安装任何修补程序后，以适当的驱动程序运行该软件以确保您的配置与已定义的安全策略保持一致。此外，请手动查看系统，因为正在使用的 **Solaris Security Toolkit** 软件的版本可能不支持由已安装的修补程序添加的新功能。
- 对系统进行实时监视以确保不会发生未经授权的行为。查看系统帐户、密码和访问模式，它们可以提供大量有关系统当前状态的信息。
- 部署和维护集中的 `syslog` 系统信息库以收集和解析 `syslog` 消息。可以通过收集和查看这些日志来获得有价值的信息。
- 制定一个全面的安全漏洞和审计策略来监视和维护系统配置。当在一段时间内维护系统处于安全的配置时，该要求尤为重要。
- 定期使用 **Solaris Security Toolkit** 软件的最新版本更新您的系统。

Solaris Security Toolkit 软件包括用作起点的默认安全性配置文件。

升级、安装和运行安全性软件

本章提供下载、升级或安装和运行 **Solaris Security Toolkit** 软件及其他与安全相关的软件的指导。其中包含将环境配置为独立或 **JumpStart** 模式以及获得支持的指导。

请按照本节提供的指导和过程以升级或安装、配置和执行软件。这些指导包括下载附加的安全性软件、帮助实例和准则。

虽然 **Solaris Security Toolkit** 软件是单机产品，然而当与下载的附加的安全性软件同时使用时最有效。该软件包括来自 **SunSolve Online** 的最新推荐的修补程序群集和安全性修补程序群集、**Solaris OS** 版本中未包含的 **Secure Shell** 软件、用于限制 **Solaris OS** 和第三方软件权限的权限和所有权修改软件，以及用于验证 **Sun** 文件和可执行文件完整性的完整性验证二进制文件。

本章包含以下任务：

- 第 31 页 “执行规划及安装前的任务”
- 第 32 页 “软件依赖性”
- 第 32 页 “确定使用何种模式”
- 第 33 页 “升级过程”
- 第 35 页 “下载安全性软件”
- 第 42 页 “定制安全性配置文件”
- 第 42 页 “安装和执行软件”
- 第 53 页 “验证系统修改”

执行规划及安装前的任务

正确的规划是成功使用 **Solaris Security Toolkit** 软件以保护系统安全的关键。有关安装软件前规划的详细信息，请参见第 2 章。

如果要在经过部署的系统上安装软件，请参见第 27 页 “执行安装前任务”，以了解有关在经过部署的系统上执行安装前任务的信息。

软件依赖性

Solaris Security Toolkit 4.2 软件依赖于 `SUNWloc` 软件包。该软件包的缺失会导致 **Solaris Security Toolkit** 发生故障。

有关所支持的 **Solaris** 操作系统版本，请参见第 **xx** 页“支持的 **Solaris OS** 版本”。

有关所支持的 **System Managements Services (SMS)** 软件版本的信息，请参见第 **xxi** 页“支持的 **SMS** 版本”。

确定使用何种模式

在 **OS** 安装过程中或者在安装结束后马上加强系统安全性，可以减少系统在不安全状态下可能遭受攻击的时间。在使用 **Solaris Security Toolkit** 软件保护系统安全之前，请配置 **Solaris Security Toolkit** 软件，使其能在您的环境中正常运行。

Solaris Security Toolkit 软件具有模块化的框架。即便当前没有使用 **JumpStart** 产品，而 **Solaris Security Toolkit** 软件框架的灵活性使您能够有效地为以后使用 **JumpStart** 版本做准备。如果正在使用 **JumpStart**，则由于 **Solaris Security Toolkit** 软件能够集成到现有的 **JumpStart** 体系结构中，因此您可从中受益。

以下几节介绍了独立模式和 **JumpStart** 模式。

独立模式

在独立模式下，**Solaris Security Toolkit** 软件可直接从 **Solaris OS shell** 提示符下运行。这种模式使您可以在那些需要安全性修改或更新但还不能中断服务以完全重新安装 **OS** 的系统上使用 **Solaris Security Toolkit** 软件。但是，只要有可能，就应该在进行保护前完全重新安装操作系统。

在安装修补程序或第三方软件后加强系统安全性时，独立模式特别有用。可以在系统上多次运行 **Solaris Security Toolkit** 软件而没有不良后果。修补程序会覆盖或修改 **Solaris Security Toolkit** 软件已经修改的文件；再次运行 **Solaris Security Toolkit** 软件，会重新执行由修补程序安装所撤消的任何安全性修改。

注 – 在生产环境中，将修补程序安装到现实环境之前，请在测试和开发环境中对其进行测试。

独立模式是尽可能快地加强已部署系统安全性的最佳选择之一。除了第 35 页“下载安全性软件”提供的下载和安装指导中的步骤之外，不需要任何专门的步骤即可将 Solaris Security Toolkit 软件集成到不带 JumpStart 的体系结构中。

JumpStart 模式

JumpStart 技术是 Sun 的基于网络的 Solaris OS 安装机制，能够在安装过程中运行 Solaris Security Toolkit 脚本。本书假设读者熟悉 JumpStart 技术并具有可用的 JumpStart 环境。有关 JumpStart 技术的更多信息，请参阅 Sun BluePrints 书籍《JumpStart Technology: Effective Use in the Solaris Operating Environment》。

Solaris Security Toolkit 4.2 软件包是可重定位的，这样可通过使用 pkgadd 命令的正确选项将该软件包安装至您所希望的任何目录。JASS_HOME_DIR 将成为 Jumpstart 服务器的基目录。

将 Solaris Security Toolkit 软件集成到 JumpStart 体系结构中只需要几个步骤。有关如何配置 JumpStart 服务器的指导，请参见第 5 章。

升级过程

本节包含的信息是关于在升级和不升级 Solaris OS 的情况下，如何将系统的 Solaris Security Toolkit 4.0 和 4.1 软件升级到 Solaris Security Toolkit 4.2 软件。通过在 Solaris 操作系统上使用 Solaris Security Toolkit 软件可以加强系统安全性。无论是从版本 4.0 还是 4.1 进行升级，过程是相同的。按照此处介绍的过程进行操作是很重要的，这可以防止您覆写以前所有的定制设置。



注意 – 每次只能安装 Solaris Security Toolkit 的一个版本。

Solaris Security Toolkit 4.2 软件中对 pkgrm 命令做了新的改进。在这一版本中，pkgrm 命令的第一步是检查发行版本中所有文件的完整性。如果存在任何不同的文件，则 pkgrm 命令会退出，并显示一个错误消息，提示系统管理员放置正确的文件或删除已修改的文件。

驱动程序放置在 Drivers 子目录下（Solaris Security Toolkit 安装在此处）。用户写入的驱动程序也放置在此处。如果定制的驱动程序与 Solaris Security Toolkit 提供的驱动程序具有不同的名称，则在使用 pkgrm 命令删除 SUNWjass 时，会同时删除 Solaris Security Toolkit 提供的驱动程序和用户修改过的驱动程序，但不会删除用户添加的定制驱动程序。



注意 – 如果某个驱动程序被修改过，则在升级前**必须**对其进行保存。请勿修改与 Solaris Security Toolkit 软件一起发行的原始文件。不要修改驱动程序文件，而是将其复制到一个新文件，然后对新文件进行修改。

▼ 升级 Solaris Security Toolkit 软件和 Solaris 操作系统

1. 按照可用的最佳操作来升级系统，即对其进行备份或使用 Solaris 升级程序。
2. 卸载以前版本的 Solaris Security Toolkit 软件。
3. 安装 Solaris Security Toolkit 4.2 软件。
4. 使用以前的 Solaris Security Toolkit 驱动程序和用户指定的驱动程序，在升级后的系统中以审计模式运行 Solaris Security Toolkit 4.2 软件。

用户指定的驱动程序必须位于 Drivers 目录中。如果确是如此，则可将其指定用于 jass-execute 或加强系统安全性运行操作。
5. 执行下列操作之一：
 - a. 如果没有错误，请转至步骤 6。
 - b. 如果在运行过程中生成错误（例如：一个非安装的运行控制脚本被修改，或者应使用 FMRI 控制某个服务），则请修正这些错误，然后重复执行步骤 4 和 5，直到不再生成错误为止。
6. 将定制的驱动程序与 secure.driver 进行比较，以确定是否应该将新的 finish 或 audit 脚本添加到定制的驱动程序。
7. 执行下列操作之一：
 - a. 如果没有缺少脚本，请转至步骤 8。
 - b. 如果缺少脚本，则请添加这些缺少的脚本，然后重复步骤 4、5、6 和 7，直到包含全部必需的脚本为止。
8. 以加强系统安全性全模式运行 Solaris Security Toolkit 4.2。
9. 以审计模式运行 Solaris Security Toolkit 4.2，并确保没有错误发生。
10. 查看系统的安全性配置和状态，以确定其是否符合安全性要求。
11. 执行下列操作之一：
 - a. 如果系统符合要求，请转至步骤 12。
 - b. 如果系统不符合要求，请更新正在使用的驱动程序，然后返回步骤 8。

12. 对系统进行完整测试，以确保系统能够提供所需的网络服务并且所有的应用程序运行完全正常。
13. 如果出现错误，请更新正在使用的驱动程序，然后返回步骤 8。
至此升级完成。

▼ 仅升级 Solaris Security Toolkit 软件

1. 卸载以前版本的 **Solaris Security Toolkit** 软件。
2. 安装 **Solaris Security Toolkit 4.2** 软件。
3. 转至第 34 页“升级 **Solaris Security Toolkit** 软件和 **Solaris** 操作系统”的步骤 4。

仅升级 Solaris OS

如果您仅打算升级 Solaris OS 并且已经安装了 **Solaris Security Toolkit 4.2** 软件（例如，从 **Solaris 8 OS** 升级到 **Solaris 10 OS**），则无需卸载 **Solaris Security Toolkit 4.2** 软件。完成对 **Solaris OS** 的升级后，请以审计模式运行 **Solaris Security Toolkit 4.2**，然后查看系统的安全性配置以确保没有错误。

下载安全性软件

加强系统安全性的第一步是将附加的安全性软件包下载到要保护的系统上。本节涵盖以下主题：

- 第 36 页 “下载 **Solaris Security Toolkit** 软件”
- 第 37 页 “下载推荐的修补程序群集软件”
- 第 38 页 “下载 **FixModes** 软件”
- 第 39 页 “下载 **OpenSSH** 软件”
- 第 40 页 “下载 **MD5** 软件”

注 – 本节介绍的软件中，**Solaris Security Toolkit** 软件、推荐的修补程序群集和安全性修补程序群集、**FixModes** 和 **message-digest 5 (MD5)** 算法软件是必需的。可以用 **Secure Shell**（可从许多供应商获得）的商业版本来替代 **OpenSSH**。在所有系统上安装并使用 **Secure Shell** 产品。如果使用的是 **Solaris 9** 或 **10 OS**，请使用其附带的 **Secure Shell (SSH)** 版本。如果使用的是 **Solaris 10 OS**，请将其附带的 `/usr/bin/digest` 命令用于 **MD5** 校验和。

下载 Solaris Security Toolkit 软件

Solaris Security Toolkit 软件以 Solaris OS 软件包格式分发。首先下载 Solaris Security Toolkit 软件，然后以独立模式将其安装在要使用 Solaris Security Toolkit 软件的服务器上，或者以 JumpStart 模式安装在 JumpStart 服务器上。

注 – 以下指导中使用的文件名未涉及版本号。请始终从网站下载所提供的最新版本。

在本指南的剩余部分中，JASS_HOME_DIR 环境变量是指 Solaris Security Toolkit 软件的根目录（默认情况下为 /opt/SUNWjass）。

▼ 下载 pkg 版本

1. 下载软件发行文件 (SUNWjass-*n.n*.pkg.tar.Z)。

源文件位于：

<http://www.sun.com/security/jass>

注 – 如果您在下载软件过程中遇到困难，请使用浏览器中的 "Save As" 选项。

2. 使用 `uncompress` 命令将软件发行文件解压缩到服务器上的一个目录中：

```
# uncompress SUNWjass-n.n.pkg.tar.Z
```

3. 使用 `tar` 命令将软件分发软件包解包：

```
# tar -xvf SUNWjass-n.n.pkg.tar
```

4. 使用 `pkgadd` 命令将软件发行文件安装到服务器上的目录中，如下所示：

```
# pkgadd -d SUNWjass-n.n.pkg SUNWjass
```

其中，*n.n* 是下载的最新版本。

执行此命令在 /opt 中创建 SUNWjass 目录。该子目录包含所有 Solaris Security Toolkit 目录和相关文件。

下载推荐的修补程序群集软件

修补程序由 Sun 发布，为 Solaris OS 提供对性能、稳定性、功能和安全性的修正。对于系统安全性最重要的是安装最新的修补程序群集。为确保系统上安装了 Solaris OS 推荐的修补程序群集和安全性修补程序群集，本节讲述了如何下载最新的修补程序群集。

注 – 在安装任何修补程序之前，请在非生产系统中或在预定的维护窗口中对其进行测试和评估。

▼ 下载推荐的修补程序群集软件

在安装修补程序群集之前，请查看每个修补程序的 README 文件和所提供的其他信息。信息通常包括安装修补程序群集之前需要知道的建议及帮助信息。

1. 从 **SunSolve OnLine Web** 站点下载最新的修补程序群集，网址为：
`http://sunsolve.sun.com`
2. 单击右侧导航栏中的 **"Patches"** 链接。
3. 单击 **"Recommended Patch Clusters"** 链接。
4. 在 **"Recommended Solaris Patch Cluster"** 框中选择适当的 **Solaris OS** 版本。
在此实例中，选择 **Solaris 10 OS**。
5. 用相关的单选按钮选择最佳的下载选项 (**HTTP** 或 **FTP**)，然后单击 **"Go"**。
在浏览器窗口中会显示一个 **"Save As"** 对话框。
6. 将文件保存到本地。
7. 将文件安全地移动至要加强安全性的系统中。
使用安全复制命令 `scp(1)`，或其他提供安全文件传输的方法。
按如下所示使用 `scp` 命令：

```
# scp 10_Recommended.zip target01:
```

8. 将文件移动到 /opt/SUNWjass/Patches 目录下并解压缩。

例如：

代码示例 3-1 将一个修补程序文件移动到 /opt/SUNWjass/Patches 目录

```
# cd /opt/SUNWjass/Patches
# mv /directory in which file was saved/10_Recommended.zip .
# unzip 10_Recommended.zip
Archive:      10_Recommended.zip
  creating:  10_Recommended/
  inflating: 10_Recommended/CLUSTER_README
  inflating: 10_Recommended/copyright
  inflating: 10_Recommended/install_cluster
[. . .]
```

在下载了附加安全性软件包并执行了 Solaris Security Toolkit 软件后，系统会自动安装修补程序群集软件。

注 - 如果没有将推荐的修补程序群集和安全性修补程序群集软件放到 /opt/SUNWjass/Patches 目录下，则当执行 Solaris Security Toolkit 软件时会显示一条警告消息。在没有修补程序群集应用时可以安全地忽略这条消息，因为这种情形通常出现在新的操作系统版本中。

下载 FixModes 软件

FixModes 是一个限制默认的 Solaris OS 目录和文件权限的软件包。限制这些权限能够有效地提高整体安全性。更多的限制性权限可以使恶意用户更难在系统中获得权限。

注 - 在 Solaris 10 OS 版本中，为了提高以前由 FixModes 软件修改的默认的对象权限，作了重大更改，因此该软件不再是必需的了。因此，install-fixmodes finish 和 audit 脚本无法用于运行 Solaris 10 OS 的系统。

▼ 下载 FixModes 软件

1. 从以下网址下载 FixModes 预编译的二进制文件：

<http://www.sun.com/security/jass>

FixModes 软件是作为预编译和压缩的软件包版本文件发行的，其格式适用于 Solaris OS 系统。文件名是 SUNBEfixm.pkg.Z。

2. 使用 `scp` 命令或者其他提供安全文件传输的方法，将文件安全地移动到要加强安全性的系统中。

按如下所示使用 `scp` 命令：

```
# scp SUNBfixm.pkg.Z target01:
```

3. 用以下命令解压缩文件 `SUNBfixm.pkg.Z` 并将其保存在 `/opt/SUNWjass/Packages` 下的 **Solaris Security Toolkit** `Packages` 目录中。

```
# uncompress SUNBfixm.pkg.Z
# mv SUNBfixm.pkg /opt/SUNWjass/Packages/
```

在下载了所有附加安全性软件包并执行了 **Solaris Security Toolkit** 软件后，会自动安装 **FixModes** 软件。

下载 OpenSSH 软件

在任何安全的环境下，需要将加密与强身份验证结合使用以保护用户交互式会话。最低限度，访问网络必须进行加密。

最常用于实行加密的工具是 **Secure Shell** 软件（**Solaris OS** 附带的版本、第三方商业版本或免费版本）。要实现 **Solaris Security Toolkit** 软件所进行的所有安全性修改，必须具有一个 **Secure Shell** 软件产品。

注 – 如果您正在使用的是 **Solaris 9** 或 **10 OS**，请使用操作系统附带的 **Secure Shell** 版本。**Secure Shell** 的该版本集成了其他 **Solaris OS** 安全功能（例如，基本安全模块 **(BSM)**）并受到 **Sun** 的支持机构的支持。

执行 **Solaris Security Toolkit** 软件禁用了系统中所有未加密的用户交互式服务和守护程序，特别是守护程序，例如：`in.telnetd`、`in.ftpd`、`in.rshd` 和 `in.rlogind`。

Secure Shell 使您可以使用 **Telnet** 和 **FTP** 访问系统。

▼ 下载 OpenSSH 软件

注 – 如果服务器当前运行 **Solaris 9** 或 **10 OS**，则可以使用附带的 **Secure Shell** 软件并跳过本节中的 **OpenSSH** 安装步骤。`install-ssh finish` 和 `audit` 脚本无法用于运行 **Solaris 10 OS** 的系统。

- 获取以下 **Sun BluePrints OnLine** 文章或 **Sun BluePrints** 书籍，并使用下载软件的指导：
 - **Sun BluePrints OnLine** 上的一篇有关如何编译与部署 **OpenSSH** 的名为 "**Building and Deploying OpenSSH on the Solaris Operating Environment**" 的文章可以在下列网址获得：
<http://www.sun.com/blueprints>
 - 可从书店购买 **Sun BluePrints** 出版物 《**Secure Shell in the Enterprise**》。
- 在下载了所有附加安全性软件包并执行了 **Solaris Security Toolkit** 软件后，会自动安装 **OpenSSH** 软件。



注意 – 请不要在需要加强安全性的系统上编译 **OpenSSH**，也不要再在需要加强安全性的系统上安装编译器。使用单独的 **Solaris OS** 系统 – 运行相同的 **Solaris OS** 版本、体系结构和模式（例如，**Solaris 8 OS**、**Sun4U™ (sun4u)** 和 **64位**）– 编译 **OpenSSH**。如果要使用 **SSH** 的商业版本，则不需要任何编译程序。目的是限制潜在入侵者对编译器的使用。但是，对于本地系统编译器安装的限制并不能对抗蓄意攻击者，因为他们仍然可以安装预编译的工具。

下载 MD5 软件

MD5 软件在要加强安全性的系统上生成 **MD5** 数字指纹。生成数字指纹后，将其与 **Sun** 公司发布的正确数字指纹相对比，以检测被未授权用户更改或隐藏在看起来安全的事物中（**特洛伊式更改**）的系统二进制文件。通过修改系统二进制文件，攻击者可利用后门访问系统；他们隐藏了自己的存在并使系统在不稳定的状态下运行。

注 – 如果服务器当前运行 **Solaris 10 OS**，则可以使用 **OS** 附带的 `/usr/bin/digest` 命令并跳过本节下面的 **MD5** 安装步骤。

▼ 下载 MD5 软件

注 – 在 **Solaris 10** 系统中，**Solaris Security Toolkit** 并不会按此过程中所述安装 **MD5** 软件并审计 **MD5** 软件的安装。运行 **Solaris 10 OS** 的系统不需要 **MD5** 软件，原因是 `digest(1M)` 命令现在包含了 **MD5** 功能。

1. 从以下网址下载 **MD5** 二进制文件：

<http://www.sun.com/security/jass>

MD5 程序以压缩的软件包版本文件形式发行。

2. 使用 `scp` 命令或其他提供安全文件传输的方法，将文件 `SUNBEmd5.pkg.Z` 安全地移动到要加强安全性的系统中。

按如下所示使用 `scp` 命令：

```
# scp SUNBEmd5.pkg.Z target01:
```

3. 使用与以下相似的命令，解压缩文件并将其移动到 `/opt/SUNWjass/Packages` 下的 **Solaris Security Toolkit Packages** 目录中。

```
# 解压缩 SUNBEmd5.pkg.Z
# mv SUNBEmd5.pkg /opt/SUNWjass/Packages/
```

将 MD5 软件保存到 `/opt/SUNWjass/Packages` 目录中后，执行 **Solaris Security Toolkit** 软件以安装 MD5 软件。

在 MD5 二进制文件安装完毕后，通过 **Solaris** 指纹数据库可以使用它们验证系统上可执行文件的完整性。有关 **Solaris** 指纹数据库的更多信息，可以从 **Sun BluePrints OnLine** 中标题为 "The Solaris Fingerprint Database — A Security Tool for Solaris Software and Files" 的文章中获得。

4. (可选) 从位于以下位置的 **Sun BluePrint Web** 站点下载并安装 **Solaris Fingerprint Database Companion** 和 **Solaris Fingerprint Database Sidekick** 软件：

<http://www.sun.com/blueprints/tools>

注 — 虽然步骤 4 被标记为可选，但在所有的操作系统中使用该步骤都是非常有益的。

安装这些可选工具并与 MD5 软件一起使用。这些工具简化了通过 MD5 校验和数据库验证系统二进制文件的过程。需要经常使用这些工具来验证安全的系统上 **Solaris OS** 二进制和文件的完整性。

这些工具及相关下载指导在 **Sun BluePrints OnLine** 中标题为 "The Solaris Fingerprint Database — A Security Tool for Solaris Software and Files" 的文章中可以找到。

需要验证下载的安全性工具的完整性。在安装和运行 **Solaris Security Toolkit** 软件及附加安全性软件前，请使用 MD5 校验和验证其完整性。在 **Solaris Security Toolkit** 下载页面上，可以获得用于该用途的 MD5 校验和。

定制安全性配置文件

很多安全性配置文件模板是作为驱动程序包含在 **Solaris Security Toolkit** 软件发行版本中的。由驱动程序实现的安全性配置文件禁用了那些不需要的服务，并启用了被 `secure.driver` 禁用的可选的安全功能。正如前面章节所述，默认的安全性配置文件和由这些驱动程序进行的更改很可能并不适用于您的系统。

在运行 **Solaris Security Toolkit** 软件之前，请查看并定制环境的默认安全性配置文件，或者开发新的文件。《**Solaris Security Toolkit 4.2 Reference Manual**》提供了定制安全性配置文件的技巧和准则。

安装和执行软件

在执行 **Solaris Security Toolkit** 软件前完成以下预备任务是很重要的。运行 **Solaris Security Toolkit** 软件时，会自动完成大部分加强安全性工作。

- 在要加强安全性的系统上或在 **JumpStart** 服务器上下载附加安全性软件和 **Solaris Security Toolkit** 软件。请参见第 35 页“下载安全性软件”。
- 将您的系统配置为独立或 **JumpStart** 模式。请参见第 32 页“确定使用何种模式”。
- 为您的环境定制 **Solaris Security Toolkit** 软件（如果需要）。
- 在安装和运行 **Solaris Security Toolkit** 软件及附加安全性软件前，需要使用 MD5 校验和验证其完整性。

可以直接从命令行或从 **JumpStart** 服务器上执行 **Solaris Security Toolkit** 软件。

有关命令行选项及其他关于执行软件的信息，请参阅以下之一：

- 第 42 页“在独立模式下执行软件”
- 第 52 页“在 **JumpStart** 模式下执行软件”

在独立模式下执行软件

代码示例 3-2 显示了在独立模式下使用命令行的范例。

代码示例 3-2 在独立模式下使用命令行的范例

```
# ./jass-execute -h  
  
usage:
```


代码示例 3-2 在独立模式下使用命令行的范例（续）

```

To apply this Toolkit to a system, using the syntax:
jass-execute [-r root_directory -p os_version ]
[ -q | -o output_file ] [ -m e-mail_address ]
[ -V [3|4] ] [ -d ] driver

To undo a previous application of the Toolkit from a system:
jass-execute -u [ -b | -f | -k ] [ -q | -o output_file ]
[ -m e-mail_address ] [ -V [3|4] ]

To audit a system against a pre-defined profile:
jass-execute -a driver [ -V [0-4] ] [ -q | -o output_file ]
[ -m e-mail_address ]

To remove saved files from a previous run of the Toolkit:
jass-execute -c [ -q | -o output_file ]
[ -m e-mail_address ] [ -V [3|4] ]

To display the history of Toolkit applications on a system:
jass-execute -H

To display the last application of the Toolkit on a system:
jass-execute -l

To display this help message:
jass-execute -h
jass-execute -?

To display version information for this program:
jass-execute -v

#
    
```

表 3-1 列出了可用的命令行选项并逐个进行说明。

表 3-1 将命令行选项与 jass-execute 一起使用

选项	描述
-a 驱动程序	确定系统与其安全性配置文件是否兼容。 请勿与 -b、-k、-f、-c、-d、-h、-H、-l、-p、-r 或 -u 选项一起使用。
-b	备份自最后一次加强安全性运行操作以来手动更改的所有文件，然后将系统恢复至其原始状态。 仅与 -u 选项一起使用。
-c	指定清除选项。删除以前运行 Solaris Security Toolkit 时保存的文件。
-d 驱动程序	指定在独立模式下运行的驱动程序。 请勿与 -a、-b、-c、-f、-h、-H、-k 或 -u 选项一起使用。

表 3-1 将命令行选项与 `jass-execute` 一起使用（续）

选项	描述
-f	在不询问是否有例外的情况下，取消加强安全性运行操作所作的更改，在执行加强安全性操作后进行手动更改的文件也不例外。 仅与 <code>-u</code> 选项一起使用。
-H	显示系统上 Solaris Security Toolkit 软件的历史。
-h -?	显示 <code>jass-execute</code> 帮助消息，其中概述了可用的选项。 单独使用。除 <code>-h -?</code> 之外，忽略所有指定的选项。
-k	在加强安全性运行操作后保存对文件所做的手动更改。 仅与 <code>-u</code> 选项一起使用。
-l	显示系统上已安装的 Solaris Security Toolkit 的最近应用。
-m 电子邮件地址	为内部支持指定电子邮件地址。
-o 输出文件	指定到输出文件的完整路径以及输出文件本身。
-p OS 版本	指定 Solaris OS 版本。格式与 <code>uname -r</code> 相同。 必须与 <code>-r</code> 根目录选项一起使用。
-q	指定静默模式。运行此命令时，不显示消息。输出结果被保存在 <code>JASS_REPOSITORY/</code> 中。
-r 根目录	指定在 <code>jass-execute</code> 运行时使用的根目录。此根目录是 <code>/</code> ，由 Solaris Security Toolkit 环境变量 <code>JASS_ROOT_DIR</code> 进行定义。可通过 <code>/</code> 保护 Solaris OS 的安全。例如，如果您希望保护一个单独 OS 目录（临时安装在 <code>/mnt</code> 中）的安全，则可以使用 <code>-r</code> 选项来指定 <code>/mnt</code> 。 必须与 <code>-p OS 版本</code> 选项一起使用。
-u	发生异常时，运行带有交互式提示（询问要采取何种措施）的撤消选项。 请勿与 <code>-a</code> 、 <code>-c</code> 、 <code>-d</code> 、 <code>-h</code> 、 <code>-l</code> 、 <code>-p</code> 、 <code>-r</code> 或 <code>-H</code> 选项一起使用。
-v 详细级别	指定审计运行操作的详细级别。存在 5 个级别 (0-4)
	0 单独行，表明通过或者没有通过审计。
	1 对于每个脚本，使用单独行表明通过还是没有通过，在所有脚本行的下面有一个大的总分行。
	2 对于每个脚本，提供了所有检查的结果。
	3 提供完整输出的多行，包括标题与标题消息。这是默认级别。
	4 多行（所有级别 3 中提供的数据）加上 <code>logDebug</code> 日志功能产生的所有项。本级别用于调试。
-v	显示本程序的版本信息。

有关在独立模式下可与 `jass-execute` 命令一起使用的选项的详细信息，请参见以下几节：

- 第 46 页 “审计选项”
- 第 46 页 “清除选项”
- 第 48 页 “显示帮助选项”
- 第 49 页 “驱动程序选项”
- 第 50 页 “电子邮件通知选项”
- 第 50 页 “执行历史记录选项”
- 第 50 页 “最新执行选项”
- 第 51 页 “输出文件选项”
- 第 51 页 “抑制显示输出选项”
- 第 52 页 “根目录选项”
- 第 52 页 “撤消选项”

有关可用的驱动程序的完整列表，请参见第 5 页 “Drivers 目录”。新的软件版本可能包含额外的驱动程序。

▼ 在独立模式下执行软件

1. 执行 `secure.driver`（或产品的专用脚本，例如 `sunfire_15k_sc-secure.driver`），如下所示：

代码示例 3-3 在独立模式下执行软件

```
# ./jass-execute -d secure.driver

[NOTE] The following prompt can be disabled by setting
JASS_NOVICE_USER to 0.
[WARN] Depending on how the Solaris Security Toolkit is configured,
it is both possible and likely that by default all remote shell
and file transfer access to this system will be disabled upon
reboot effectively locking out any user without console access to
the system.

Are you sure that you want to continue? (YES/NO) [NO]
Y

[NOTE] Executing driver, secure.driver

=====
secure.driver: Driver started.
=====

=====
Solaris Security Toolkit Version: 4.2.0
Node name:                ufudu
Zone name:                 global
Host ID:                   8085816e
```

代码示例 3-3 在独立模式下执行软件（续）

```
Host address:                10.8.31.115
MAC address:                 8:0:20:85:81:6e
OS version:                 5.10
Date:                       Tue Jul 5 16:28:24 EST 2005
=====
[...]
```

有关可用的驱动程序的完整列表，请参见第 5 页“Drivers 目录”。新的软件版本可能包含额外的驱动程序。

2. 在系统上运行 Solaris Security Toolkit 软件后，请重新引导系统以实现更改。

在加强安全性过程中，需要对客户机的配置做大量更改。这些更改可能包括禁用服务启动脚本，禁用服务选项和通过修补程序安装新的二进制文件或库。只有客户机重新启动后，这些更改才可能被启用。

3. 重新引导系统后，检查更改的正确性和完整性。

请参见第 53 页“验证系统修改”。

4. 如果出现任何错误，请进行修正并再次在独立模式下运行 Solaris Security Toolkit 软件。

审计选项

通过 `-a` 选项，Solaris Security Toolkit 软件能够进行审计运行，以确定系统是否与其安全性配置文件相兼容。此运行不仅验证系统文件修改是否仍然起作用，还验证之前禁用的过程是否在运行或者删除的软件包是否已重新安装。有关此功能的更多信息，请参见第 6 章。

审计系统的安全性配置文件的命令行用法摘要：

```
# jass-execute -a 驱动程序 [ -v [0-4] ] [ -q | -o 输出文件 ]
[ -m 电子邮件地址 ]
```

清除选项

`-c` 选项删除在先前运行 Solaris Security Toolkit 过程中保存的文件。您可以将抑制显示输出 (`-q`)、输出 (`-o`)、邮件 (`-m`) 和详细 (`-V`) 选项与清除选项一起使用。

代码示例 3-4 显示了使用 -c 选项的实例，其产生的输出与以下内容类似：

代码示例 3-4 -c 选项输出范例

```
# bin/jass-execute -c
Executing driver, clean.driver

Please select Solaris Security Toolkit runs to clean:
1. July 15, 2005 at 11:41:02 (/var/opt/SUNWjass/run/20050715114102)
2. July 15, 2005 at 11:44:03 (/var/opt/SUNWjass/run/20050715114403)
Choice ('q' to exit)? 2
[NOTE] Cleaning previous run from /var/opt/SUNWjass/run/20050715114403

=====
clean.driver: Driver started.
=====

=====
Toolkit Version: 4.2.0
Node name:      sstzone
Zone name:     sstzone
Host ID:       80cb346c
Host address:  10.8.28.45
MAC address:   8:0:20:cb:34:6c
OS version:    5.10
Date:         Fri Jul 15 11:44:58 PDT 2005

=====
clean.driver: Performing CLEANUP of /var/opt/SUNWjass/run/20050715114403.
=====

=====
clean.driver: Driver finished.
=====

=====
[SUMMARY] Results Summary for CLEAN run of clean.driver
[SUMMARY] The run completed with a total of 1 script run.
[SUMMARY] There were Failures in 0 Scripts
[SUMMARY] There were Errors in 0 Scripts
[SUMMARY] There were Warnings in 0 Scripts
[SUMMARY] There was a Note in 1 Script
[SUMMARY] Notes Scripts listed in:
           /var/opt/SUNWjass/run/20050715114403/jass-clean-script-notes.txt
=====
```

显示帮助选项

-h 选项显示 jass-execute 帮助信息，提供了可用选项的概述。

-h 选项产生的输出与以下内容相似：

代码示例 3-5 -h 选项输出范例

```
# ./jass-execute -h
To apply this Toolkit to a system, using the syntax:
    jass-execute [-r root_directory -p os_version ]
                [-q | -o output_file ] [ -m e-mail_address ]
                [-V [3|4] ] [ -d ] driver

To undo a previous application of the Toolkit from a system:
    jass-execute -u [ -b | -f | -k ] [ -q | -o output_file ]
                [ -m e-mail_address ] [ -V [3|4] ]

To audit a system against a pre-defined profile:
    jass-execute -a driver [ -V [0-4] ] [ -q | -o output_file ]
                [ -m e-mail_address ]

To remove saved files from a previous run of the Toolkit:
    jass-execute -c [ -q | -o output_file ]
                [ -m e-mail_address ] [ -V [3|4] ]

To display the history of Toolkit applications on a system:
    jass-execute -H

To display the last application of the Toolkit on a system:
    jass-execute -l

To display this help message:
    jass-execute -h
    jass-execute -?

To display version information for this program:
    jass-execute -v

Note that just the driver name should be specified when using the
'-d' or '-a' options. A path need not be specified as the script
is assumed to exist in the Drivers directory.

The '-u' undo option is mutually exclusive with the '-d' and '-a'
options. The default undo behavior is to ask the user what to do if
a file to be restored has been modified as the checksum is
incorrect.

The -u option can be combined with the '-k', '-b', or '-f' to
```

代码示例 3-5 -h 选项输出范例（续）

```
override the default interactive behavior. The use of one of these options is required when run in quiet mode ('-q').
```

```
The '-k' option can be used to always keep the current file and backup if checksum is incorrect. The 'b' can be used to backup the current file and restore original if the checksum is incorrect. The 'f' option will always overwrite the original if the checksum is incorrect, without saving the modified original.
```

驱动程序选项

-d 驱动程序选项指定在独立模式下运行的驱动程序。

必须用 -d 选项指定驱动程序。Solaris Security Toolkit 软件将 Drivers/ 附加在所添加脚本的名称之前。因此只需在命令行中输入脚本名称。

注 - 请勿将 -d 选项与 -a、-b、-c、-f、-H、-h、-k 或 -u 选项一起使用。

使用了 -d 驱动程序选项的 jass-execute 加强安全性运行操作产生的输出类似于以下内容：

代码示例 3-6 -d 驱动程序选项输出范例

```
# ./jass-execute -d secure.driver
[...]
[NOTE] Executing driver, secure.driver

=====
secure.driver: Driver started.
=====

Solaris Security Toolkit Version: 4.2.0
Node name:                        ufudu
Zone name:                        global
Host ID:                          8085816e
Host address:                      10.8.31.115
MAC address:                       8:0:20:85:81:6e
OS version:                        5.10
Date:                              Tue Jul 5 16:28:24 EST 2005
=====
[...]
```

电子邮件通知选项

-m **电子邮件地址**选项提供了一种机制，通过该机制，**Solaris Security Toolkit** 软件可在运行完成后将独立审计、清除、加强系统安全性和撤消运行操作的输出用电子邮件自动发送出去。电子邮件报告是使用其他选项在系统中生成的日志和由 **Solaris Security Toolkit** 软件创建的本地日志的补充。

使用电子邮件选项调用 `sunfire_15k_sc-config.driver` 的 **Solaris Security Toolkit** 运行类似于以下内容：

```
# ./jass-execute -m root -d sunfire_15k_sc-config.driver
[...]
```

执行历史记录选项

-H 选项提供一种确定 **Solaris Security Toolkit** 软件在系统中已运行多少次的简单机制。该选项将列出所有的运行，无论其撤消与否。

-H 选项产生的输出类似于以下内容：

代码示例 3-7 -H 选项输出范例

```
# ./jass-execute -H
Note: This information is only applicable for applications of
      the Solaris Security Toolkit starting with version 0.3.

The following is a listing of the applications of the Solaris
Security Toolkit on this system. This list is provided in
reverse chronological order:

1.   June 31, 2004 at 12:20:19 (20040631122019) (UNDONE)
2.   June 31, 2004 at 12:10:29 (20040631121029)
3.   June 31, 2004 at 12:04:15 (20040631120415)
```

此输出表明，**Solaris Security Toolkit** 软件已在系统上运行了三次，且最后一次运行被撤消。

最新执行选项

-l 选项提供了确定最近一次运行的机制。这通常是 -H 选项列出的最后一次运行。

-l 选项产生的输出类似于以下内容：

代码示例 3-8 -l 选项输出范例

```
# ./jass-execute -l

Note: This information is only applicable for applications of
      the Solaris Security Toolkit starting with version 4.2.0.

The last application of the Solaris Security Toolkit was:

1.   June 31, 2005 at 12:20:19 (20040631122019) (UNDONE)
```

输出文件选项

-o 输出文件选项将 jass-execute 的控制台输出重定向到一个单独的输出文件。您可以为输出文件指定完全适合的路径名。

此选项不影响 JASS_REPOSITORY 目录内的日志。在较慢的终端连接上运行时，此选项特别有帮助。根据所指定的详细级别，运行 Solaris Security Toolkit 可能会生成大量的输出。

此选项可与 -a、-d 或 -u 选项一起使用。

-o 选项产生的输出类似于以下内容：

代码示例 3-9 -o 选项输出范例

```
# ./jass-execute -o /var/tmp/root/jass-output.txt -d secure.driver
[NOTE] Executing driver, secure.driver
[NOTE] Recording output to /var/tmp/root/jass-output.txt
```

抑制显示输出选项

在加强安全性运行操作中，-q 选项可禁止 Solaris Security Toolkit 输出到控制台。

此选项不影响 JASS_REPOSITORY 目录内的日志。与 -o 选项类似，此选项特别有助于通过 cron 作业或较慢的网络连接来运行 Solaris Security Toolkit 软件。

此选项可与 -a、-c、-d 或 -u 选项一起使用。

-q 选项产生的输出类似于以下内容：

代码示例 3-10 -q 选项输出范例

```
# ./jass-execute -q -d secure.driver
[NOTE] Executing driver, secure.driver
```

根目录选项

-r 根目录选项用于指定在运行 `jass-execute` 时使用的根目录。使用 -r 选项还需要使用 -p 选项指定平台 (OS) 版本。-p 选项的格式与 `uname -r` 产生的格式相同。

此根目录是 /，由 **Solaris Security Toolkit** 环境变量 `JASS_ROOT_DIR` 进行定义。可通过 / 保护 **Solaris OS** 的安全。例如，如果要保护一个独立的 OS 目录，请临时在 /mnt 下装入，然后使用 -r 选项指定 /mnt。则会将所有脚本应用到该 OS 映像。

撤消选项

通过 -u 选项，**Solaris Security Toolkit** 软件可以在加强安全性过程中撤消系统修改。每个 `finish` 脚本都可以通过 -u 选项来撤消。此外，**Solaris Security Toolkit** 的撤消功能与每次运行时所产生的校验和紧密相关。有关此功能的更多信息，请参见第 4 章。

有三个选项可与 -u 选项一起使用：

- -b (备份) 选项，备份自最后一次加强安全性运行操作以来手动更改的所有文件，然后将系统恢复至其原始状态。
- -f (强制) 选项，在不询问是否有例外的情况下，取消加强安全性运行操作所作的更改，在执行加强安全性操作后进行手动更改的文件也不例外。
- -k (保留) 选项，保留自最后一次执行加强安全性运行操作以来所作的手动更改。

撤消命令的命令行用法摘要：

```
# jass-execute -u [ -b | -f | -k ] [ -q | -o 输出文件 ]
[ -m 电子邮件地址 ] [ -v [3|4] ]
```

在 JumpStart 模式下执行软件

JumpStart 模式由插入到 JumpStart 服务器上 `rules` 文件中的 **Solaris Security Toolkit** 驱动程序控制。

如果没有配置环境以使用 JumpStart 模式，请参阅第 5 章。

有关 JumpStart 技术的更多信息，请参阅 Sun BluePrints 书籍《JumpStart Technology: Effective Use in the Solaris Operating Environment》。

▼ 在 JumpStart 模式下执行软件

要在 JumpStart 模式下执行 Solaris Security Toolkit 软件，必须将其集成到 JumpStart 环境中并作为和 JumpStart 安装相关的 **finish** 脚本的一部分进行调用。有关如何将 Solaris Security Toolkit 软件集成到您环境中的信息，请参见第 5 章。

1. 完成对驱动程序的所有必要修改后，请使用 **JumpStart** 基础结构安装客户机。
在客户机的 ok 提示符中使用以下命令完成该任务。

```
ok> boot net - install
```

一旦安装完成，JumpStart 软件就会重新引导系统。

系统的配置应该正确。在加强安全性过程中，需要对客户机的配置做大量更改。这些更改包括禁用服务启动脚本、禁用服务选项和禁止通过修补程序安装新库。只有客户机重新启动后，这些更改才可能生效。

2. 重新引导系统后，检查更改的正确性和完整性。
请参见第 53 页“验证系统修改”。
3. 如果出现错误，请修正它们并重新安装客户机的 OS。

验证系统修改

重新引导系统后，按照以下几节所述对修改的正确性和完整性进行验证。

对服务执行 QA 检查

保证系统安全的重大挑战之一是，确定必须使哪些 OS 服务处于启用状态以使系统正常运行。由于要直接使用 Solaris OS 服务（例如，用于登录系统的 **Secure Shell**），因此可能需要它们。或者间接使用它们，例如使用第三方软件管理工具的图形用户界面 (GUI) 的远程过程调用 RPC 守护进程。

在运行 **Solaris Security Toolkit** 软件之前，应该确定这些要求中的大部分。（参见第 18 页“确定应用程序和服务要求”。）然而，**唯一**确定的机制是安装并保护系统安全，然后通过质量保证 (QA) 测试对其所要求的功能进行完全测试。在加强系统安全性后，要对所有新部署的系统正确地执行 QA 计划。类似的，对于要加强安全性的经过部署的系统，需要进行完全的测试以确保其具备了所有要求的和所期望的功能。

如果 QA 过程发现了任何差异，请执行以下步骤：

1. 根据第 2 章中的准则，确定问题出现区域。
2. 验证应用程序在已修改的配置中运行。
3. 撤消 **Solaris Security Toolkit** 运行。
4. 根据问题的解决方案，修改安全性配置文件（驱动程序）。
5. 再次运行 **Solaris Security Toolkit** 软件。

最终结果应为一个安全性配置文件，它可在系统上运行而不对任何所需的功能产生负面影响。

对配置进行安全性评估

在验证系统可执行所有需要的功能时，还要对安全性配置做出评估，以确定系统是否达到需要的安全级别。根据在系统上执行的加强安全性及最小化系统操作，这种评估可能涉及不同的方面。

至少应该从以下几方面检查系统的配置：

- 确保安装了所有适当的安全性修补程序和推荐的修补程序。
- 验证当前是否**仅**运行必需的和相关的进程，并且以正确的参数运行。
- 确保当前**仅**运行必需的守护程序，并且以适当的参数运行。
- 进行本地检查（例如，`netstat -a`）和使用端口扫描程序（例如 **Nmap**，它可以确定网络接口上可用的端口）进行远程检查，以验证系统中是否**仅**开放了必需的端口。
- 如果系统是最小化的，请确保**仅**安装了必需的 **Solaris OS** 软件包。

对于新构建的并已保护安全的系统，至少应该考虑到这种检查。在加强传统系统的安全性时，需要检查底层的 OS 以确定是否执行了未授权的修改。对于这种完整性检查，最好在只读模式下装入系统的文件系统，然后从一个已知 OS 实例上运行完整性检查软件。在 **Sun BluePrints OnLine** 中标题为 "**The Solaris Fingerprint Database - A Security Tool for Solaris Software and Files**" 的文章所介绍的工具有在这些情况下非常有用。

验证安全性配置文件

在保护系统安全并且已经验证了所需的服务和功能后，请使用审计功能以确保正确且完全地应用了安全性配置文件。此任务非常关键，原因有两个。第一，要确保按照要求加强了系统的安全性。第二，要确保为系统定义的安全性配置文件正确地反映在了 **Solaris Security Toolkit** 配置中。由于配置信息用于在系统的整个部署生命周期中维护系统的安全性配置文件，因此这种检查十分重要。

有关审计功能的更多信息，请参见第 6 章。

执行安装后的任务

如果在一台经过部署的系统上安装软件，请参见第 28 页“执行安装后的任务”，以了解有关在经过部署的系统上执行安装后任务的信息。

第4章

取消系统更改

本章提供的信息和步骤可用于取消（撤消）**Solaris Security Toolkit** 软件在加强安全性运行过程中所做的更改。此选项提供一种自动机制，通过该机制可使系统返回到**Solaris Security Toolkit** 加强安全性操作或一系列操作之前的状态。

本章包含以下主题：

- 第 57 页 “了解如何记录并取消更改”
- 第 58 页 “撤消系统更改的要求”
- 第 59 页 “定制脚本以撤消更改”
- 第 59 页 “检查手动更改的文件”
- 第 60 页 “使用具有撤消功能的选项”
- 第 63 页 “撤消系统更改”

了解如何记录并取消更改

每次 **Solaris Security Toolkit** 加强安全性操作都在 `JASS_REPOSITORY` 下创建一个运行目录。这些目录的名称基于运行的日期和时间。除了在屏幕上显示输出之外，**Solaris Security Toolkit** 软件还会在目录下创建一系列文件以跟踪更改并记录其操作。

存储在目录中的文件跟踪对系统进行的修改，并使撤消功能生效。



注意 – 管理员不应修改 `JASS_REPOSITORY` 目录中文件的内容。修改这些文件会破坏文件内容，且当使用撤消功能时，会引起不可预知的错误或系统崩溃。

当在 **JumpStart** 或者独立模式下使用 **Solaris Security Toolkit** 软件加强系统的安全性时，软件将系统更改记录在 `JASS_REPOSITORY/jass-manifest.txt` 文件中。此文件列出了 `undo` 功能用于取消更改的操作。该文件包含有关通过 **Solaris Security Toolkit** 软件加强安全性操作的信息，包括创建、复制、移动或删除的文件。另外，此文件还可包含取消更复杂的更改（例如软件包安装）时所需的条目（标准条目和定制条目）。每次加强安全性操作都会创建一个单独的 `jass-manifest.tx` 文件。

注 – Solaris Security Toolkit 软件的 `undo` 功能仅能取消那些在清单文件中具有相应条目的更改。

`undo` 运行操作会仔细检查那些在 Solaris Security Toolkit 运行过程中生成并保存在 `JASS_REPOSITORY` 中的清单文件。根据您是否使用了备份、强制或保留选项，该运行操作会将备份文件恢复到它们最初的位置。有关备份、强制和保留选项的更多信息，请参见以下内容：

- 第 61 页 “备份选项”
- 第 61 页 “强制选项”
- 第 62 页 “保留选项”

如果在加强系统安全性过程中没有备份文件，则 `JASS_SAVE_BACKUP` 变量在 `user.init` 文件中被定义为 0 时或使用 `-c` 选项时，撤消操作功能不可用。有关更多信息，请参见第 58 页 “撤消系统更改的要求”。

撤消 Solaris Security Toolkit 操作并不会删除相关的目录。而是在 `JASS_REPOSITORY` 目录下创建两个文件：`jass-undo-log.txt` 和 `reverse-jass-manifest.txt`。随后再次执行 `jass-execute -u` 时，不会列出已撤消的操作。加强安全性操作只能撤消一次。

撤消系统更改的要求

在使用 Solaris Security Toolkit 软件的撤消功能时，请注意以下限制和要求。

- 在 Solaris Security Toolkit 版本 0.3 到 4.2 中，可以在独立模式或者 JumpStart 模式下对已开始进行的运行操作使用撤消功能。但是，**仅**可以在独立模式下撤消更改。在 JumpStart 安装过程中**不能**使用撤消功能。
- 如果选择了 Solaris Security Toolkit 选项，不创建备份文件；则无论在 JumpStart 模式还是在独立模式下，撤消功能均不可用。将 `JASS_SAVE_BACKUP` 参数设为 0，可禁止创建备份文件副本。
- 一个运行操作**只能**撤消一次。
- 如果您要开发一个新的 `finish` 脚本，请确保使用 Solaris Security Toolkit 框架功能。您必须创建一个匹配的 `audit` 脚本并使用 `add_to_manifest` 功能为清单文件添加条目。否则，Solaris Security Toolkit 无法了解您定制的开发。
- 在任何情况下都**不要**修改 `JASS_REPOSITORY` 目录的内容。修改文件会破坏文件内容，且当使用撤消功能时，会引起不可预知的错误或系统崩溃。

定制脚本以撤消更改

Solaris Security Toolkit 框架为设计和构建 **finish** 脚本提供了灵活性。该框架允许扩展 **Solaris Security Toolkit** 软件的功能以更好地满足组织的要求，同时也能帮助您在配置的生命周期内改善对系统配置的管理。

当定制脚本时，了解所采取的操作如何影响撤消功能是非常重要的。为了简化定制脚本，助手功能可对清单文件做适当的更改。（**undo** 功能根据清单文件的内容取消加强系统安全性运行操作。）在多数情况下，这些助手功能提供了为您机构定制脚本所需的操作。

有关助手功能的列表和使用它们的信息，请参阅《**Solaris Security Toolkit 4.2 Reference Manual**》。使用这些助手功能代替它们对应的系统命令，从而使撤消运行操作能够参考清单文件中的相关条目。

在某些情况下，要执行的功能可能不具有助手功能。在这些情况下，请使用名为 **add_to_manifest** 的特殊功能。使用此功能可将条目手动插入到清单文件中，而不必调用助手功能。请慎重使用此特殊功能，以保护系统和 **Solaris Security Toolkit** 信息库的完整性。例如，可使用此特殊功能添加非 **Sun** 的 **pkg** 格式的软件包。在此实例中，需要通知 **undo** 功能如何删除在加强安全性运行操作过程中以其他格式添加的软件包。

利用助手功能和特殊的 **add_to_manifest** 功能，**Solaris Security Toolkit** 软件提供了一种简单灵活的方法，可定制脚本，并可撤消运行操作对系统进行的更改。

如果更改 **finish** 脚本时，没有使用这些功能，则 **Solaris Security Toolkit** 无法了解已经发生的更改。因此，必须手动撤消清单文件没有相关记录的任何更改。

在另一个实例中，在更改系统中的文件之前，应首先保存文件的原始版本。在 **Solaris Security Toolkit** 软件环境外，用户通常执行 **/usr/bin/cp** 命令以完成该任务。然而，在 **Solaris Security Toolkit** 软件环境下，如果直接使用此命令，**Solaris Security Toolkit** 软件无从了解需要创建一个清单条目。因此，请使用 **backup_file** 助手功能，而不要使用 **cp** 命令。该功能以 **JASS_SUFFIX** 为后缀保存了原始文件的副本，并添加了一个清单条目以通知 **Solaris Security Toolkit** 软件已复制了文件的副本。同时此功能还使文件校验和开始计算。文件校验和由撤消功能以及 **jass-check-sum** 命令使用。

检查手动更改的文件

虽然使用 **jass-execute -u** 命令可以自动检查在加强安全性运行操作后手动更改的文件，但有时使用 **jass-check-sum** 命令列出并检查已经更改的文件也很有帮助。

此命令可检查 JASS_REPOSITORY 目录的内容并对清单文件中列出的所有文件执行校验和，以确定自加强安全性运行操作（执行并保存校验和）以来，哪些文件进行了更改。在强制运行撤消操作之前进行检查，该操作可提供有价值的信息，从而节省数小时不必要的错误诊断时间。

以下为一个输出实例。

代码示例 4-1 手动更改的文件输出范例

# ./jass-check-sum			
File Name	Saved CkSum	Current CkSum	
/etc/inet/inetd.conf	1643619259:6883	2801102257:6879	
/etc/logadm.conf	2362963540:1042	640364414:1071	
/etc/default/inetd	3677377803:719	2078997873:720	

该输出表明加强安全性运行操作结束后更改了三个文件。

使用具有撤消功能的选项

本节介绍了 `jass-execute -u` 命令和在执行撤消操作时可使用的选项。

注 – 不能在撤消功能中使用 `-c`、`-d`、`-a`、`-h`、`-l` 或 `-H` 选项。在静默模式下运行撤消操作时，必须提供 `-b`、`-k` 或 `-f` 选项。

`jass-execute -u` 命令是执行 `undo` 运行操作的标准方法。此命令会自动发现上次执行加强安全性操作以来手动进行的任何更改。如果 **Solaris Security Toolkit** 软件发现了上次执行加强安全性操作以来手动更改的文件，系统会提示您选择以下响应操作之一：

1. 在恢复原始的文件（加强安全性运行操作前存在的文件）之前，请备份最新的文件。
2. 保留最新的文件，且**不要**恢复原始文件。
3. 强制覆写手动更改的任何文件（可能会导致数据丢失）并恢复原始文件。
4. 在恢复原始的文件（加强安全性运行操作前存在的文件）之前，务必备份最新的文件。
5. 务必保留最新的文件，且**不要**恢复原始文件。
6. 务必强制覆写手动更改的所有文件（可能会导致数据丢失）并恢复原始文件。

如果您希望定义撤消命令应如何处理执行加强安全性操作后被修改过的文件，请在执行撤消命令时使用备份 (`-b`)、保留 (`-k`) 或强制 (`-f`) 选项。

表 4-1 列出了可与撤消操作一起使用的命令行选项。有关每个选项的详细信息，请参见以下几节。

表 4-1 将命令行选项与撤消命令一起使用

选项	描述
-b	备份自最后一次加强安全性运行操作以来手动更改的所有文件，然后将系统恢复至其原始状态。
-f	在不询问是否有例外的情况下，取消加强安全性运行操作所作的更改，在执行加强安全性操作后进行手动更改的文件也不例外。
-k	在加强安全性运行操作后保存对文件所做的手动更改。
-m	将输出发送到一个电子邮件地址。
-o	定向输出到一个文件。
-q	防止将输出显示到屏幕上。又称为抑制显示输出选项。输出存储在 JASS_REPOSITORY/jass-undo-log.txt 文件中。
-V	指定撤消操作运行的详细级别。

备份选项

-b 选项自动备份自上次加强安全性运行操作以来手动更改的任何文件，然后将文件恢复到加强安全性运行操作前的原始状态。为实现手动更改，**必须**将恢复文件与备份文件作对比，并手动调整它们之间的不同。使用此选项备份文件的实例用法如下。

```
/etc/motd.BACKUP.JASS_SUFFIX
```

强制选项

-f 选项取消加强安全性运行操作过程中未发生异常情况的更改，也包括在加强安全性运行操作之后手动更改的文件。该撤消操作运行**不会**将所保存的文件校验和与现有文件版本的校验和进行比较。因此，如果在加强安全性运行操作后手动更改了文件，则在执行具有强制选项的撤消操作后，文件更改会被覆盖或者丢失。

您可能需要在撤消运行操作完成后再次进行手动更改。此外，可能有必要根据所做的更改类型调整文件组之间的差异。

注 - 为避免发生这些问题，请使用前面提到的 `jass-check-sum` 命令或 `-b` 命令行选项。

保留选项

-k 选项自动保留加强安全性运行操作后所做的手动更改而不恢复原始文件。-k 选项可搜索文件中的任何不匹配内容，生成通知并进行记录，但不用原始文件覆盖现有文件。只有那些保存了有效校验和的更改才会被取消。

此选项并非没有缺点。例如，如果使用 **finish** 脚本修改了一部分文件之后又手动修改了这些文件，则该选项会导致系统不一致。

以 **finish** 脚本 `remove-unneeded-accounts.fin` 为例。该脚本修改了系统中的 `/etc/passwd` 和 `/etc/shadow` 文件。如果用户在加强安全性运行操作完成后手动更改了密码，则与 `/etc/shadow` 文件相关的校验和将与 **Solaris Security Toolkit** 软件保存的值不匹配。所导致的结果就是，如果使用保留选项，则只有 `/etc/passwd` 文件会被复制回它的原始状态。而 `/etc/shadow` 文件会保持它的当前形式。这两个文件不再一致。

输出文件选项

-o 到输出文件的完整路径选项将 `jass-execute` 的控制台输出重定向到一个单独的输出文件。

此选项不影响 `JASS_REPOSITORY` 目录中的日志。终端连接缓慢时，可使用该选项，因为 **Solaris Security Toolkit** 撤消操作会产生大量的输出。

抑制显示输出选项

注 — 在静默模式下运行撤消操作时，必须提供 -b、-k 或 -f 选项。

-q 选项可阻止 **Solaris Security Toolkit** 软件将输出显示到屏幕上。此选项不影响 `JASS_REPOSITORY` 目录中的日志。与 -o 选项类似，此选项特别有助于通过 `cron` 作业或较慢的网络连接来运行 **Solaris Security Toolkit** 软件。

电子邮件通知选项

-m 电子邮件地址选项指示 **Solaris Security Toolkit** 软件将一个已完成运行操作的副本发送到指定的电子邮件地址。使用其他选项除了可在系统上生成日志外，还可生成电子邮件报告。

撤消系统更改

有时必须取消在一个或多个 **Solaris Security Toolkit** 加强安全性运行操作过程中所做的更改。如果发现在加强安全性运行操作时所做的更改对系统产生了负面影响，请撤消这些更改。

例如，如果在加强安全性运行操作后发现禁用了一项所需的服务，例如 **Solaris Volume Manager (SVM)**，请执行以下操作：

1. 撤消加强安全性操作。
2. 创建一个定制的驱动程序。
有关定制驱动程序的指导，请参阅《**Solaris Security Toolkit 4.2 Reference Manual**》第 4 章中的 "Customizing Drivers"。
3. 通过 `JASS_SVCS_ENABLE` 环境变量启用您希望使用的 SVM 服务。
有关使用 `JASS_SVCS_ENABLE` 的指导，请参阅《**Solaris Security Toolkit 4.2 Reference Manual**》第 7 章中的 "JASS_SVCS_ENABLE"。
4. 重复加强安全性操作。

本节提供了在一个或者多个加强安全性运行操作的过程中取消更改的指导。请注意，对于有效地取消加强安全性运行操作有一些限制和要求。请参阅第 58 页“撤消系统更改的要求”。

▼ 撤消 Solaris Security Toolkit 运行操作

1. 备份并重新引导系统。
在每次撤消运行前备份并重新引导系统，以确保能够返回或恢复到一个已知的工作状态。
2. 确定在使用 `jass-execute -u` 命令时使用哪些选项。
请参见第 60 页“使用具有撤消功能的选项”。
以下指导假设您正在使用 `jass-execute -u` 命令。

3. 要使用标准 `-u` 选项撤消一个或多个加强安全性运行操作，请在 `JASS_HOME_DIR/bin` 中输入以下命令：

```
# ./jass-execute -u
```

Solaris Security Toolkit 软件通过查找 `JASS_REPOSITORY` 中的所有清单文件收集有关每次加强安全性运行操作的信息。如果清单文件为空或者根本不存在，则认为没有可撤消的更改并忽略该运行操作。此外，如果一个名为 `jass-undo-log.txt` 的文件与清单文件存在于同一目录下，则认为已运行了取消操作，从而忽略该运行操作。收集过程完成之后，会显示结果。以下是一个输出实例。

代码示例 4-2 可撤消的运行操作的输出范例

```
# ./jass-execute -u
[NOTE] Executing driver, undo.driver
Please select a JASS run to restore through:
1. January 24, 2003 at 13:57:27
   (/var/opt/SUNWjass/run/20030124135727)
2. January 24, 2003 at 13:44:18
   (/var/opt/SUNWjass/run/20030124134418)
3. January 24, 2003 at 13:42:45
   (/var/opt/SUNWjass/run/20030124134245)
4. January 24, 2003 at 12:57:30
   (/var/opt/SUNWjass/run/20030124125730)

Choice? ('q' to exit)?
```

在本实例中，存在四个独立的加强安全性运行操作。这些运行操作使系统发生更改并且尚未被撤消。加强安全性运行操作列表通常按时间从近到远列出。列表中的第一项为最近执行的加强安全性运行操作。

4. 检查输出以确定需要撤消哪些运行操作，然后输入相应的数字。

对于所选的项，Solaris Security Toolkit 软件取消每个索引号等于或小于所选值的运行操作。即撤消运行操作将按照与原先所做更改相反的顺序来撤消这些更改，从最近的加强安全性运行操作开始一直到所选的运行操作。使用前面的实例作为指导，如果选择了运行操作 3，则撤消运行操作首先取消运行操作 1 的更改，然后取消运行操作 2 的更改，最后取消运行操作 3 的更改。

代码示例 4-3 示了撤消运行操作处理两个清单文件条目时生成的输出。

代码示例 4-3 撤消运行操作处理多个清单文件条目的输出范例

```
[...]  
  
===== undo.driver: Performing UNDO of  
//var/opt/SUNWjass/run/20050715145837.  
===== undo.driver: Undoing Finish Script: update-cron.allow.fin  
===== undo.driver: Undoing operation COPY.  
cp -p /etc/cron.d/cron.allow.JASS.20050715145906  
/etc/cron.d/cron.allow  
rm -f /etc/cron.d/cron.allow.JASS.20050715145906  
  
[NOTE] Removing a Solaris Security Toolkit-created file.  
rm -f /etc/cron.d/cron.allow  
  
[...]
```

在本实例中，**Solaris Security Toolkit** 软件撤消了一个复制操作并删除了加强安全性运行过程中所添加的一个文件。撤消运行操作的输出记录了用于恢复系统的实际命令，在需要对系统配置进行错误诊断时可以清楚地理解此过程并用作参考。

对于自上次执行加强安全性运行操作以来进行过修改的文件，如果 **Solaris Security Toolkit** 的检查是成功的，则撤消运行操作将会继续进行，直到处理完所有的运行操作和相应的清单文件并取消了更改为止。

Solaris Security Toolkit 软件除了通过查找所有位于 `JASS_REPOSITORY` 中的清单文件以收集有关每次加强安全性运行的信息之外，它还会执行以下操作：

- a. 比较每个经过修改的文件的校验和。
 - b. 为校验和文件中任何不匹配的情况生成一个通知并进行记录。
 - c. 询问您希望对这些文件采取什么操作。
5. 如果撤消运行操作发现异常（在加强安全性运行操作后更改了文件），请输入这些选项之一。

注 – Solaris Security Toolkit 软件记录了您对某个特定异常文件的备份、保留和强制选择；当该文件再次成为某个撤消运行操作的异常时，您无需对该文件做出选择。

以下是一个显示了异常和处理异常选择的实例。

代码示例 4-4 撤消异常的输出范例

```
[...]  
  
=====undo.driver: Undoing Finish Script: enable-process-accounting.fin=====
```

```
[NOTE] Undoing operation COPY.  
[WARN] Checksum of current file does not match the saved value.  
[WARN] filename = /var/spool/cron/crontabs/adm  
[WARN] current = db27341e3e1f0f27d371d2e13e6f47ce  
[WARN] saved = a7f95face84325cddc23ec66d59374b0
```

```
Select your course of action:  
1. Backup - Save the current file, BEFORE restoring original.  
2. Keep - Keep the current file, making NO changes.  
3. Force - Ignore manual changes, and OVERWRITE current file.
```

```
NOTE: The following additional options are applied to this and ALL  
subsequent files:  
4. ALWAYS Backup.  
5. ALWAYS Keep.  
6. ALWAYS Force.
```

```
Enter 1, 2, 3, 4, 5, or 6:
```


在本实例中，如果选择第 1 项，则会显示以下输出。

代码示例 4-5 在撤消过程中选择备份选项的输出范例

```
Enter 1, 2, 3, 4, 5, or 6: 1

[WARN] Creating backup copies of some files may cause unintended
effects.
[WARN] This is particularly true of /etc/hostname.[interface]
files as well as crontab files in /var/spool/cron/crontabs.

[NOTE] BACKUP specified, creating backup copy of
/var/spool/cron/crontabs/adm.
[NOTE] File to be backed up is from an undo operation.
[NOTE] Copying /var/spool/cron/crontabs/adm to
/var/spool/cron/crontabs/adm.BACKUP.JASS.20050715151817
cp -p /var/spool/cron/crontabs.JASS/adm.JASS.20050715151719
/var/spool/cron/crontabs/adm
rm -f /var/spool/cron/crontabs.JASS/adm.JASS.20050715151719

[NOTE] Undoing operation COPY.
cp -p /var/spool/cron/crontabs.JASS/root.JASS.20050715151717
/var/spool/cron/crontabs/root
rm -f /var/spool/cron/crontabs.JASS/root.JASS.20050715151717

[NOTE] Undoing operation MAKE DIRECTORY.
rmdir /var/spool/cron/crontabs.JASS

[NOTE] Undoing operation SYMBOLIC LINK.
rm -f /etc/rc3.d/S22acct

[NOTE] Undoing operation SYMBOLIC LINK.
rm -f /etc/rc0.d/K22acct
```

如果选择第 4 项，则会显示以下输出。

代码示例 4-6 在撤消过程中选择备份选项的输出范例

```
Enter 1, 2, 3, 4, 5, or 6: 4
[NOTE] Always do BACKUP selected.  Overriding JASS_UNDO_TYPE with
BACKUP.

[WARN] Creating backup copies of some files may cause unintended
effects.
[WARN] This is particularly true of /etc/hostname.[interface]
files as well as crontab files in /var/spool/cron/crontabs.

[NOTE] BACKUP specified, creating backup copy of
/var/spool/cron/crontabs/adm.
[NOTE] File to be backed up is from an undo operation.
[NOTE] Copying /var/spool/cron/crontabs/adm to
/var/spool/cron/crontabs/adm.BACKUP.JASS.20050715152126
cp -p /var/spool/cron/crontabs.JASS/adm.JASS.20050715151953
/var/spool/cron/crontabs/adm
rm -f /var/spool/cron/crontabs.JASS/adm.JASS.20050715151953

[NOTE] Undoing operation COPY.
[WARN] Checksum of current file does not match the saved value.
[WARN]     filename = /var/spool/cron/crontabs/root
[WARN]     current  = 741af21a62ea7a9e7abe6ba04855aa76
[WARN]     saved    = bcf180f45c65ceff3bf61012cb2b4982
[WARN] Creating backup copies of some files may cause unintended
effects.
[WARN] This is particularly true of /etc/hostname.[interface]
files as well as crontab files in /var/spool/cron/crontabs.
[NOTE] BACKUP specified, creating backup copy of
/var/spool/cron/crontabs/root.
[NOTE] File to be backed up is from an undo operation.
[NOTE] Copying /var/spool/cron/crontabs/root to
/var/spool/cron/crontabs/root.BACKUP.JASS.20050715152127
cp -p /var/spool/cron/crontabs.JASS/root.JASS.20050715151951
/var/spool/cron/crontabs/root
rm -f /var/spool/cron/crontabs.JASS/root.JASS.20050715151951

[NOTE] Undoing operation MAKE DIRECTORY.
rmdir /var/spool/cron/crontabs.JASS

[NOTE] Undoing operation SYMBOLIC LINK.
rm -f /etc/rc3.d/S22acct

[NOTE] Undoing operation SYMBOLIC LINK.
rm -f /etc/rc0.d/K22acct
```

在 **Solaris Security Toolkit** 完成了选定的撤消操作后，您应该检查标记为执行加强安全性操作之后更改过的文件，并对系统进行必要的修改。在手动检查完这些更改之前，请勿重新引导系统，原因是这样可能会使系统处于不一致的状态。

注 – 在实例中，以新的文件名保存修改过的文件：

`/etc/.login.BACKUP.JASS.20050715151817`。撤消运行操作完成之后，将该文件与 `/etc/.login` 相比较以确定是否需要进一步进行统一。

6. 在继续之前解决异常情况。

7. 解决完异常情况后，请重新引导系统。

要使对 **Solaris OS** 配置所做的修改生效，重新引导系统是必要的。



注意 – **Solaris Security Toolkit** 在 **JumpStart** 模式下运行时，会设置 `root` 密码。如果稍后执行了撤消操作，则 `root` 密码会恢复到先前无密码的设置。这意味着完全没有密码，任何人都可以登录到超级用户帐户。在执行撤消操作之后，请记住使用 `passwd(1)` 命令重置 `root` 密码。在系统处于这种状态时，**Solaris Security Toolkit 4.2** 软件还会显示一条警告消息。

配置和管理 JumpStart 服务器

本章提供了配置和管理 JumpStart 服务器以使用 Solaris Security Toolkit 软件的信息。JumpStart 技术是 Sun 基于网络的 Solaris OS 安装机制，能够在安装过程中运行 Solaris Security Toolkit 软件。

Solaris Security Toolkit 的 JumpStart 模式基于 JumpStart 技术，且对于 Solaris OS 2.1 以后的产品都是可用的。JumpStart 技术可帮助您使 Solaris OS 和系统软件安装完全自动化，实现系统的正确性和标准化，以进行复杂地管理。它提供了一种方法，可满足快速安装和部署系统的要求。

在系统安全领域使用 JumpStart 技术的优点是明显的。通过将 JumpStart 技术与 Solaris Security Toolkit 软件结合使用，可以在 Solaris OS 自动安装过程中保护系统安全。此做法有助于确保在系统安装时已使系统安全标准化并进行了处理。JumpStart Enterprise Toolkit (JET) 软件可帮助进行基于 JumpStart 的安装并包含支持使用 Solaris Security Toolkit 加强系统安全性的模块，要获取该软件，请访问 Sun Software Download，网址为：

<http://www.sun.com/download/>

有关 JumpStart 技术的更多信息，请参阅 Sun BluePrints 书籍《JumpStart Technology: Effective Use in the Solaris Operating Environment》。

本章包含以下主题：

- 第 72 页 “配置 JumpStart 服务器和环境”
- 第 74 页 “使用 JumpStart 配置文件模板”
- 第 75 页 “添加和删除客户机”

配置 JumpStart 服务器和环境

为在 JumpStart 环境中进行使用，请将 `/opt/SUNWjass` 中的 Solaris Security Toolkit 源代码（用于 pkg 下载）安装到 JumpStart 服务器的基目录下。默认目录是 JumpStart 服务器上的 `/jumpstart`。完成该任务后，`JASS_HOME_DIR` 成为 Jumpstart 服务器上的基目录。

本节假设读者熟悉 JumpStart 技术并已经有一个可用的 JumpStart 运行环境。

将 Solaris Security Toolkit 软件集成到 JumpStart 体系结构中只需要几个步骤。

▼ 配置 JumpStart 模式

1. 将 Solaris Security Toolkit 源代码安装到 JumpStart 服务器的根目录下。

可以将 Solaris Security Toolkit 安装到 `JASS_REPOSITORY`，在此处为 `/jumpstart`，如以下实例所示：

```
# pwd
/opt/SUNWjass
# pkgadd -R /jumpstart -d . SUNWjass
```

Solaris Security Toolkit 软件通常安装在 JumpStart 服务器上的 `SI_CONFIG_DIR` 目录中，通常也会是 `JASS_HOME_DIR`。

2. 如果对 Solaris 2.5.1 OS `sysidcfg` 文件做了修改，则也要对 `JASS_HOME_DIR/Sysidcfg/Solaris_2.5.1` 目录中的文件做相应修改。

如果使用的是 Solaris 2.5.1 OS，则不能直接使用

`JASS_HOME_DIR/Sysidcfg/Solaris_2.5.1` 中的 `sysidcfg` 文件，因为 Solaris 的该版本仅支持 `SI_CONFIG_DIR` 中的而不是独立子目录中的 `sysidcfg` 文件。为解决 Solaris 2.5.1 OS 的这种限制，Solaris Security Toolkit 软件具有了 `SI_CONFIG_DIR/sysidcfg`，它链接到 `JASS_HOME_DIR/Sysidcfg/Solaris_2.5.1/sysidcfg` 文件。

3. 使用以下命令将 `JASS_HOME_DIR/Drivers/user.init.SAMPLE` 复制为 `JASS_HOME_DIR/Drivers/user.init`。

```
# pwd
/jumpstart/opt/SUNWjass/Drivers
# cp user.init.SAMPLE user.init
```

- 如果您希望在安装 **JumpStart** 的过程中，将 **Solaris Security Toolkit** 软件包安装到目标系统，则必须将软件包置于在 `user.init` 文件中定义的 `JASS_PACKAGE_MOUNT` 目录下。例如：

```
# cp /路径/到/SUNWjass.pkg JASS_HOME_DIR/Packages
```

- 如果在使用多宿主的 **JumpStart** 服务器时遇到问题，请修改 `JASS_PACKAGE_MOUNT` 和 `JASS_PATCH_MOUNT` 中的两个条目，以将路径纠正为 `JASS_HOME_DIR/Patches` 和 `JASS_HOME_DIR/Packages` 目录。
- 如果要在 `SI_CONFIG_DIR` 子目录（例如，`SI_CONFIG_DIR/path/to/JASS`）中安装 **Solaris Security Toolkit** 软件，请向 `user.init` 文件中添加以下内容：

```
if [ -z "${JASS_HOME_DIR}" ]; then
    if [ "${JASS_STANDALONE}" = 0 ]; then
        JASS_HOME_DIR="${SI_CONFIG_DIR}/path/to/JASS"
    fi
fi
export JASS_HOME_DIR
```

- 选择或创建一个 **Solaris Security Toolkit** 驱动程序（例如，默认的 `secure.driver`）。
 - 如果要使用 `hardening.driver` 和 `config.driver` 中列出的所有脚本，请向 `rules` 文件中添加 `Drivers/secure.driver` 路径。
 - 如果仅使用所选脚本，请复制这些文件，然后修改其副本。有关复制和修改驱动程序的指导，请参阅《**Solaris Security Toolkit 4.2 Reference Manual**》的第 4 章中的 "Customizing Drivers"。



注意 – 请勿修改 **Solaris Security Toolkit** 软件中附带的原始脚本。为允许 **Solaris Security Toolkit** 软件新版本有效的迁移，请单独保存原始文件和定制的文件。

- 完成驱动程序后，在 `rules` 文件中创建适当的条目。

该条目与以下内容相似：

```
hostname imbulu - Profiles/core.profile Drivers/secure-abc.driver
```

要成功地将 **Solaris Security Toolkit** 软件集成到现有的 **JumpStart** 环境中，可能还需要做另一项修改。

- 如果使用 **Solaris Security Toolkit** 软件提供的 `sysidcfg` 文件以使 **JumpStart** 客户机安装自动化，请检查它们的适用性。

如果在解析 `sysidcfg` 文件时 **JumpStart** 服务器遇到问题，则会忽略整个文件的内容。

完成本节所有的配置步骤之后，可以在客户机上使用 **JumpStart** 技术安装 **Solaris OS**，并可在安装过程中成功地加强或者最小化 **OS** 的安全性。

使用 **JumpStart** 配置文件模板

JumpStart 配置文件模板是只能在 **JumpStart** 模式中使用的文件。在 **Sun BluePrints** 书籍《**JumpStart Technology: Effective Use in the Solaris Operating Environment**》中介绍了配置文件中必需的和可选的内容。

可使用 **JumpStart** 配置文件模板作为范例，在此基础上针对您的站点要求进行修改。检查配置文件以确定需要进行哪些更改（如果有），以应用于您的环境。

复制配置文件，然后在您的环境中修改其副本。不要更改原始文件，因为更新 **Solaris Security Toolkit** 软件时可能会覆盖定制的内容。

以下是 **Solaris Security Toolkit** 软件中包含的 **JumpStart** 配置文件。

- `core.profile`
- `end-user.profile`
- `developer.profile`
- `entire-distribution.profile`
- `oem.profile`
- `minimal-SunFire_Domain*.profile`

以下几小节介绍这些配置文件。

`core.profile`

该 **JumpStart** 配置文件安装最小的 **Solaris OS** 群集 (**SUNWCreq**)。除了指定包含根分区与交换分区的磁盘分区外，没有做任何其他的配置修改。

`end-user.profile`

该 **JumpStart** 配置文件安装 **End User Solaris OS** 群集 (**SUNWCuser**) 和使进程统计正常运行所需的两个 **Solaris OS** 软件包。另外，将磁盘分区定义为仅包含根分区和交换分区。

developer.profile

该 JumpStart 配置文件安装 Developer Solaris OS 群集 (SUNWCprog) 和使进程统计正常运行所需的两个 Solaris OS 软件包。与在 core.profile 中定义的一样，除了 Solaris OS 群集之外唯一需要定义的配置是为了使磁盘分区包含根分区和交换分区。

entire-distribution.profile

该 JumpStart 配置文件安装 Entire Distribution Solaris OS 群集 (SUNWca11)。和其他配置文件一样，将磁盘分区定义为包含根分区和交换分区。

oem.profile

该 JumpStart 配置文件安装 OEM Solaris OS 群集 (SUNWCxall)。本群集是 Entire Distribution 群集的父群集，它安装 OEM 提供的软件。

minimal-SunFire_Domain*.profile

注 – 仅在运行 Solaris OS 版本 8 和 9 的系统中使用这些配置文件。

以下所有配置文件均基于 Sun BluePrints OnLine 文章 "Minimizing Domains for Sun Fire V1280, 12K, and 15K Systems"。以下 JumpStart 配置文件与该文章提及的相同。

- minimal-SunFire_Domain-Apps-Solaris8.profile
- minimal-SunFire_Domain-Apps-Solaris9.profile
- minimal-SunFire_Domain-NoX-Solaris8.profile
- minimal-SunFire_Domain-NoX-Solaris9.profile
- minimal-SunFire_Domain-X-Solaris8.profile
- minimal-SunFire_Domain-X-Solaris9.profile

添加和删除客户机

add-client 和 rm-client 脚本用于配置服务器，这样服务器可以使用 JumpStart 软件来执行基于网络的客户机安装。这两个脚本位于 JASS_HOME_DIR/bin 目录下。JumpStart 模式由插入到 JumpStart 服务器上 rules 文件中的 Solaris Security Toolkit 驱动程序控制。

如果没有为使用 JumpStart 模式而配置环境，请参见第 72 页“配置 JumpStart 服务器和环境”。

对于基于 SPARC 的系统，add-client 命令将安装 JumpStart 客户机和 Solaris Security Toolkit 所需的配置信息。该命令从 JumpStart 服务器执行。

对于 x86/x64 系统（需要动态主机配置协议 (DHCP) 客户机），您需要使用 Solaris（安装）介质附带的 add_install_client 脚本。

add-client 脚本

为简化从 JumpStart 服务器添加客户机的过程，请使用包含在 Solaris Security Toolkit 软件中的该脚本。以下段落介绍了这些命令和选项，但没有介绍底层的 JumpStart 技术。有关 JumpStart 技术的信息，请参阅 Sun BluePrints 书籍《JumpStart Technology: Effective Use in the Solaris Operating Environment》。

add-client 脚本是 add_install_client 命令的外壳，它接受以下参数。

add-client 命令摘要：

```
# add-client -c 客户机 -i 服务器 -m 客户机类 -o 客户机 OS -s sysidcfg
```

表 5-1 介绍了 add-client 命令的有效输入。

表 5-1 JumpStart add-client 命令

选项	描述
-c 客户机	JumpStart 客户机的可解析的主机名。
-h -?	显示使用信息。使用时不能带有任何其他选项。忽略所有其他选项。
-i 服务器	用于该 JumpStart 客户机的 JumpStart 服务器接口的 IP 地址或可解析的主机名。如果没有指定值，则会显示本地主机上可用接口的列表。
-m 客户机类	JumpStart 客户机的计算机类型。该值与 uname -m 命令的输出格式相同。
-o 客户机 OS	可以在 JASS_HOME_DIR/OS 目录下获得的将要安装在客户机上的 Solaris OS 的版本。如果没有指定值，则会显示 JASS_HOME_DIR/OS 目录中可用的 Solaris OS 版本的列表。
-s sysidcfg	备用目录的可选路径名，此目录包含一个用于系统识别和配置的 sysidcfg 文件。默认情况下，该值被设置为 JASS_HOME_DIR/Sysidcfg/Solaris 版本/目录，其中 Solaris 版本是从您使用的必要参数 -o 中提取的。如果要指定一个可选的路径名，请使用到 JASS_HOME_DIR 目录的相对路径名。仅指定到 sysidcfg 文件的路径。
-v	本程序的版本信息。忽略所有其他选项。

要使用默认值添加名为 `eng1` 的 `JumpStart` 客户机，您可以执行以下操作：

```
# /opt/SUNWjass/bin/add-client -c eng1 -m sun4u
Selecting default operating system, Solaris 版本.
Selecting default system interface, IP 地址.
cleaning up preexisting install client "eng1"
removing eng1 from bootparams
updating /etc/bootparams
```

要使用 `Solaris 9 OS (12/03)` 和 `-s sysidcfg` 选项，将名为 `eng1` 的 `JumpStart` 客户机添加到名为 `jumpserve1` 的 `JumpStart` 服务器，您可以执行以下操作：

```
# /opt/SUNWjass/bin/add-client -c eng1 -i jumpserve1 -m sun4u -o
Solaris_9_2003-12 -s Hosts/alpha
cleaning up preexisting install client "eng1"
removing eng1 from bootparams
updating /etc/bootparams
```

rm-client 脚本

该脚本包含在 `Solaris Security Toolkit` 软件中，可简化从 `JumpStart` 服务器中删除客户机的操作。请使用包含在 `Solaris Security Toolkit` 软件中的该脚本。以下段落介绍了这些命令和选项，但没有介绍底层的 `JumpStart` 技术。有关 `JumpStart` 技术的信息，请参阅《`JumpStart Technology: Effective Use in the Solaris Operating Environment`》。

与 `add-client` 相似，`rm-client` 脚本是 `rm_install_client` 命令的外壳：

用法实例：`rm-client [-c] 客户机`

其中，`客户机`是 `JumpStart` 客户机的可解析的主机名。

表 5-2 介绍了 `rm-client` 命令的有效输入。

表 5-2 `JumpStart rm-client` 命令

选项	描述
<code>-c 客户机</code>	<code>JumpStart</code> 客户机的可解析的主机名。
<code>-h -?</code>	显示使用信息。使用时不能带有任何其他选项。忽略所有其他选项。
<code>-v</code>	本程序的版本信息。忽略所有其他选项。

要删除一台名为 `eng1` 的 **JumpStart** 客户机，请使用以下 `rm-client` 命令：

```
# ./rm-client -c eng1
removing eng1 from bootparams
```

第6章

审计系统安全性

本章介绍如何使用 **Solaris Security Toolkit** 软件审计（验证）系统的安全性。可在加强安全性后使用本章的信息和过程维护已建立的安全性配置文件。对于经过部署的系统，在加强安全性之前需要使用本章中的信息对安全性进行评估。

注 – 本章和本书中使用术语**审计**来定义 **Solaris Security Toolkit** 软件通过与预定义的安全性配置文件做比较以自动验证安全性状态的过程。在本出版物中使用该术语并不能保证在使用审计选项后系统是完全安全的。

本章包含以下主题：

- 第 79 页 “维护安全性”
- 第 80 页 “执行加强操作之前检查安全性”
- 第 80 页 “定制安全性审计”
- 第 81 页 “审计安全性的准备工作”
- 第 82 页 “使用选项和控制审计输出”
- 第 89 页 “进行安全性审计”

维护安全性

维护安全性是一个进行中的过程，必须要定期检查及访问。维护系统安全性需要警惕性，因为任何系统默认的安全性配置会随着时间的逝去而愈加趋向公开。（有关维护安全性的更多信息，请参见第 29 页 “维护系统安全”）

通过使用指定的安全性配置文件确定其兼容级别，**Solaris Security Toolkit** 提供了一种自动审计系统安全性状态的方法。

注 – 本方法仅能用于使用 `jass-execute -a` 命令的独立模式下，而不能在 **JumpStart** 安装过程中使用。

定期审计系统的安全性状态，既可以通过手动方式也可以通过自动方式（例如，通过 cron 作业或 rc 脚本）。例如，在加强一个全新安装系统的安全性之后，请在五天后执行 **Solaris Security Toolkit** 软件审计命令（`jass-execute -a 驱动程序名称`）以确定系统安全性是否已更改（与安全性配置文件所定义的状态不同）。

审计安全性的频繁程度取决于环境的安全策略与紧急程度。有些用户每小时、每天或者每月进行一次审计运行操作。有些用户每小时运行一次最小扫描（仅检查有限数量），每天运行一次完整扫描（进行全部检查）。除了例行的其他审计之外，在每次重新引导系统之后，都必须检查系统的安全性状态。

审计所有重要组件以维护已部署系统的安全性状态。如果不定期审计安全性状态，则由于平均信息量或者修改会不知不觉或恶意地更改所需的安全性状态，导致配置随之变化。如果不进行定期检查，则不会发现这些更改并采取相应的纠正措施。这将导致系统的安全性降低，更容易受攻击。

除了定期审计之外，在系统升级、修补程序安装与其他的重要系统配置发生更改后也需要进行审计。

执行加强操作之前检查安全性

在某些情况下，您可能会发现在加强已部署系统的安全性之前检查其安全性状态是有用的。例如，如果接管之前由他人管理的已部署系统，则应检测系统以了解其状态，如果必要的话可以使它们与其他系统所使用的相同安全性配置文件相兼容。

定制安全性审计

审计选项为评估系统状态提供了一种高度灵活与可扩展的机制。像加强安全性脚本一样，也可以定制 **audit** 脚本的操作。例如，可以定制环境变量、定制框架与助手功能、添加新的检查和为审计框架添加功能。

大部分用户会发现标准的或产品专用的 **audit** 脚本可以作为自己定制 **audit** 脚本的模板。对于这种情形，可以通过驱动程序、**finish** 脚本、环境变量与文件模板定制 **audit** 脚本。只需很少的工作即可完成这些定制更改，且无需修改代码。进行审计时，**Solaris Security Toolkit** 软件可自动了解加强安全性运行操作中所做的任何更改。

有些用户需要创建全新所有权的或站点专用的驱动程序和脚本。使用模板和范例作为编写新驱动程序和脚本代码的准则。请注意，使用审计选项时 **Solaris Security Toolkit** 软件无法自动了解站点专用的驱动程序、**finish** 脚本、变量与功能。例如，如果添加了一个名为 `abcc-nj-secure.driver` 的站点专用驱动程序，且该驱动程序包含站点专用

的 **finish** 脚本 `abcc-nj-install-foo.fin`，则需要创建一个站点专用的 **audit** 脚本 `abcc-nj-install-foo.aud`。同样地，如果仅启动 **audit** 脚本，则应当创建与之匹配的 **finish** 脚本。

有时候，需要添加 **Solaris Security Toolkit** 软件未提供的检查或功能。对于这种情形，请向 **audit** 脚本中添加检查或者新的功能。（您或许会在相应的 **finish** 脚本中做相关更改。）在通过 `user.run` 文件添加和修改代码时需要格外慎重以避免引入错误及故障。

要定制或创建新的驱动程序、脚本、变量和功能，请参阅《**Solaris Security Toolkit 4.2 Reference Manual**》。

例如，可能需要添加一个 **Solaris Security Toolkit** 软件没有安装的修补程序。则既可以扩展标准的或者产品专用的模板，也可以创建自己的模板。如果您要创建自己的模板，请首先创建一个 **finish** 脚本来添加修补程序，然后创建相应的 **audit** 脚本来检查修补程序的安装。如果您要使用现有的 **finish** (`.fin`) 和 **audit** (`.aud`) 脚本作为自己的模板，则必须将这两个脚本复制到新建的且名称唯一的文件中。

审计安全性的准备工作

为使用本章的指导和准则，您需要一个安全性配置文件。有关开发与实现安全性配置文件的信息，请参见第 2 章。

许多安全性配置文件包含在 **Solaris Security Toolkit** 附带的驱动程序内。正如本书前面所述，默认的安全性配置文件和由其进行的更改很可能并不适用于您的系统。通常情况下，所实现的安全性配置文件是用于安全性的 "high-water" 标记。这表示它们能够禁用不需要的服务并启用被 `secure.driver` 禁用的可选安全性功能。

许多 **Solaris Security Toolkit** 软件用户发现标准的和产品专用的安全性配置文件适用于他们的环境。如果您遇到这种情形，请确定与安全性状态最接近的安全性配置文件，并将其用于评估和加强系统的安全性。

为您的环境检查并定制安全性配置文件或者开发新的文件。《**Solaris Security Toolkit 4.2 Reference Manual**》提供了定制安全性配置文件的技巧和准则。此方法提供了为您的组织定制的安全性状态，并且可以将安全性评估过程中返回的故障错误减至最少。例如，如果需要启用 **Telnet**，则可以定制安全性配置文件，使其在安全性评估过程中不会将 **Telnet** 视为安全漏洞。这样，一个使用 **Telnet** 和 **Kerbero**（用于验证和加密）的站点不会将使用 **Telnet** 视为安全漏洞。

使用选项和控制审计输出

本节介绍了执行审计运行操作的可用选项和控制输出的选项。本节包括以下内容：

- 第 82 页 “命令行选项”
- 第 85 页 “标题与消息输出”
- 第 88 页 “主机名、脚本名称和时间戳输出”

命令行选项

以下的命令行摘要显示了如何使用安全性配置文件审计系统的安全性：

```
# jass-execute -a 驱动程序 [ -v [0-4]] [ -q | -o 输出文件 ] [ -m 电子邮件地址 ]
```

当执行 Solaris Security Toolkit 软件审计命令时，可以使用表 6-1 中所列的选项。

表 6-1 将命令行选项与审计命令结合使用

选项	描述
-a 驱动程序	确定系统与其安全性配置文件是否兼容。
-m 电子邮件地址	为内部支持指定电子邮件地址。
-o 输出文件	为 Solaris Security Toolkit 运行的输出指定一个文件名。
-q	指定静默模式。运行该命令时，不显示消息。输出结果被保存在 JASS_REPOSITORY/ 中。
-v 详细级别	指定审计运行操作的详细级别 (0-4)。

有关 jass-execute -a 命令可用选项的详细信息，请参见以下几节：

- 第 82 页 “显示帮助选项”
- 第 83 页 “电子邮件通知选项”
- 第 84 页 “输出文件选项”
- 第 84 页 “抑制显示输出选项”
- 第 84 页 “详细选项”

显示帮助选项

-h 选项显示 jass-execute 帮助消息，提供了可用选项的概述。

-h 选项产生的输出与以下内容相似:

代码示例 6-1 -h 选项输出范例

```
# ./jass-execute -h

To apply this Toolkit to a system, using the syntax:
  jass-execute [-r root_directory -p os_version ]
  [ -q | -o output_file ] [ -m e-mail_address ]
  [ -V [3|4] ] [ -d ] driver

To undo a previous application of the Toolkit from a system:
  jass-execute -u [ -b | -f | -k ] [ -q | -o output_file ]
  [ -m e-mail_address ] [ -V [3|4] ]

To audit a system against a pre-defined profile:
  jass-execute -a driver [ -V [0-4] ] [ -q | -o output_file ]
  [ -m e-mail_address ]

To display the history of Toolkit applications on a system:
  jass-execute -H

To display the last application of the Toolkit on a system:
  jass-execute -l

To display this help message:
  jass-execute -h
  jass-execute -?

To display version information for this program:
  jass-execute -v
```

电子邮件通知选项

-m 电子邮件地址选项提供了一种机制, 通过该机制, Solaris Security Toolkit 软件的审计运行操作完成后可使用电子邮件将输出自动发送。使用其他选项除了可在系统上生成日志外, 还可生成电子邮件报告。

使用电子邮件选项调用 sunfire_15k_sc-config.driver 的 Solaris Security Toolkit 运行类似于以下内容:

```
# ./jass-execute -m root -a sunfire_15k_sc-config.driver
[...]
```

输出文件选项

`-o` 输出文件选项将 `jass-execute` 运行的控制台输出重新定向到一个单独的文件（输出文件）。

此选项不影响 `JASS_REPOSITORY` 目录内的日志。此选项对于在较慢的终端连接上运行特别有帮助，因为 **Solaris Security Toolkit** 运行会产生大量输出。

此选项可与 `-d`、`-u` 或 `-a` 选项一起使用。

`-o` 选项产生的输出类似于以下内容：

代码示例 6-2 `-o` 选项输出范例

```
# ./jass-execute -o jass-output.txt -a secure.driver
[NOTE] Executing driver, secure.driver
[NOTE] Recording output to jass-output.txt
#
```

抑制显示输出选项

`-q` 选项禁止 **Solaris Security Toolkit** 在执行加强安全性运行操作时输出到标准输入/输出 (`stdio`) 流。

此选项不影响 `JASS_REPOSITORY` 目录内的日志。与 `-o` 选项类似，此选项特别有助于通过 `cron` 作业或较慢的网络连接来运行 **Solaris Security Toolkit** 软件。

此选项可与 `-d`、`-u` 或 `-a` 选项一起使用。

`-q` 选项产生的输出类似于以下内容：

代码示例 6-3 `-q` 选项输出范例

```
# ./jass-execute -q -a secure.driver
[NOTE] Executing driver, secure.driver
```

详细选项

`-v` 选项指定了审计运行操作的详细级别。此选项仅可用于审计。详细级别为显示审计运行操作结果提供了高度灵活的方式。例如，如果要审计 100 台计算机，您需要将每台计算机的输出限制到单独一行，以便容易确定哪些计算机通过审计而哪些计算机没有通过审计。然后对于没有通过审计的计算机，可能需要进行一个能够提供扩展输出的审计以将注意力集中到有问题的区域。

-v 选项控制五个详细级别（0 到 4）。每个递增的级别都提供了附加细节，可用它来更加完整地理解哪些检查通过而哪些检查没有通过。表 6-2 介绍了详细级别。

表 6-2 审计详细级别

级别	输出
0	单独行，表明通过或者没有通过审计。
1	对于每个脚本，用单独一行表明通过还是没有通过审计，在所有脚本行的下面有一个重要的总分数行。
2	对于每个脚本，提供了所有检查的结果。
3	提供完整输出的多行，包括标题与标题消息。这是默认级别。
4	多行（所有级别 3 中提供的数据）加上 logDebug 日志功能产生的所有项。本级别用于调试。

注 - jass-execute -v 命令默认的详细级别为 3。

有关审计冗余级别的完整描述，请参阅 jass-execute 手册页或《Solaris Security Toolkit 4.2 Reference Manual》中第 7 章的 "JASS_VERBOSITY"。

标题与消息输出

可以配置 Solaris Security Toolkit 审计选项以报告或者忽略标题和消息。JASS_LOG_BANNER 变量不能用于详细级别 0-2。这些输出选项适用于详细级别 3 和 4。例如，您可能要从输出中忽略掉通过的消息（JASS_LOG_SUCCESS 变量）以便报告并且仅将注意力集中到没有通过的消息（JASS_LOG_FAILURE 变量）上。

表 6-3 列出了能够通过日志变量控制的标题和消息。（有关日志变量的详细信息，请参阅《Solaris Security Toolkit 4.2 Reference Manual》中第 7 章的 "JASS_LOG_BANNER"。有关冗余级别的更多信息，请参阅 jass-execute 手册页或《Solaris Security Toolkit 4.2 Reference Manual》中第 7 章的 "JASS_VERBOSITY"。）如果日志变量设置为 0，则不会产生这种类型消息的任何输出。相反，如果日志变量设置为 1，那么会显示出消息。每个变量默认的操作是显示输出。表 6-3 介绍了日志变量。

表 6-3 在审计输出中显示标题和消息

日志变量	日志前缀	描述
JASS_LOG_BANNER	所有标题输出	此参数控制标题消息的显示。这些消息周围通常都是由等号 (=) 或者破折号 (-) 组成的分隔符。
JASS_LOG_ERROR	[ERR]	此参数控制错误消息的显示。如果设置为 0，则不会产生任何错误消息。
JASS_LOG_FAILURE	[FAIL]	此参数控制失败消息的显示。如果设置为 0，则不会产生任何失败消息。
JASS_LOG_NOTICE	[NOTE]	此参数控制通知消息的显示。如果设置为 0，则不会产生任何通知消息。
JASS_LOG_SUCCESS	[PASS]	此参数控制成功或者通过状态消息的显示。如果设置为 0，则不会产生任何成功消息。
JASS_LOG_SUMMARY	[SUMMARY]	此参数控制摘要消息的显示。如果设置为 0，则不会产生任何摘要消息。
JASS_LOG_WARNING	[WARN]	此参数控制警告消息的显示。如果设置为 0，则不会产生任何警告消息。

仅当需要查看特定消息时使用这些选项是非常有用的。通过设置这些选项，能够最小化输出，并仍然将注意力集中到您认为重要的地方。例如，除了 JASS_LOG_FAILURE（保留其默认值 1）之外，将所有日志变量设置为 0，则审计只会报告由 logFailure 功能产生的错误。

代码示例 6-4 仅报告审计失败的输出范例

```
# JASS_LOG_FAILURE=1
# export JASS_LOG_FAILURE
# JASS_LOG_BANNER=0
# JASS_LOG_ERROR=0
# JASS_LOG_NOTICE=0
# JASS_LOG_SUCCESS=0
# JASS_LOG_SUMMARY=0
# JASS_LOG_WARNING=0
# export JASS_LOG_BANNER JASS_LOG_ERROR
# export JASS_LOG_NOTICE JASS_LOG_SUCCESS
# export JASS_LOG_WARNING
# bin/jass-execute -a abc.driver -V2
update-at-deny      [FAIL] User adm is not listed in /etc/cron.d/at.deny.
update-at-deny      [FAIL] User zz999999 is not listed in /etc/cron.d/at.deny.
update-at-deny      [FAIL] User gdm is not listed in /etc/cron.d/at.deny.
update-at-deny      [FAIL] User lp is not listed in /etc/cron.d/at.deny.
update-at-deny      [FAIL] User nobody4 is not listed in /etc/cron.d/at.deny.
update-at-deny      [FAIL] User root is not listed in /etc/cron.d/at.deny.
update-at-deny      [FAIL] User smmsp is not listed in /etc/cron.d/at.deny.
update-at-deny      [FAIL] User sys is not listed in /etc/cron.d/at.deny.
```

代码示例 6-4 仅报告审计失败的输出范例（续）

```

update-at-deny [FAIL] User uucp is not listed in /etc/cron.d/at.deny.
update-at-deny [FAIL] User webservd is not listed in /etc/cron.d/at.deny.
update-at-deny [FAIL] Script Total: 10 Errors
update-inetd-conf [FAIL] Service svc:/network/telnet:default was enabled.
update-inetd-conf [FAIL] Service svc:/network/ftp:default was enabled.
update-inetd-conf [FAIL] Service svc:/network/finger:default was enabled.
update-inetd-conf [FAIL] Service svc:/network/login:rlogin was enabled.
update-inetd-conf [FAIL] Service svc:/network/shell:default was enabled.
update-inetd-conf [FAIL] Service svc:/network/login:eklogin was enabled.
update-inetd-conf [FAIL] Service svc:/network/login:klogin was enabled.
update-inetd-conf [FAIL] Service svc:/network/shell:kshell was enabled.
update-inetd-conf [FAIL] Service svc:/application/font/stfsloader:default was
enabled.
update-inetd-conf [FAIL] Service svc:/network/security/kttk_warn:default was
enabled.
update-inetd-conf [FAIL] Service svc:/network/rpc/smsserver:default was
enabled.
update-inetd-conf [FAIL] Service svc:/network/rpc/rstat:default was enabled.
update-inetd-conf [FAIL] Service svc:/network/rpc/rusers:default was enabled.
update-inetd-conf [FAIL] Service svc:/network/nfs/rquota:default was enabled.
update-inetd-conf [FAIL] Service svc:/network/rpc/gss:default was enabled.
update-inetd-conf [FAIL] Service 100235 is enabled in /etc/inet/inetd.conf.
update-inetd-conf [FAIL] Service 100083 is enabled in /etc/inet/inetd.conf.
update-inetd-conf [FAIL] Service 100068 is enabled in /etc/inet/inetd.conf.
update-inetd-conf [FAIL] Script Total: 18 Errors
abc.driver [FAIL] Driver Total: 28 Errors
abc.driver [FAIL] Grand Total: 28 Errors
#

```

主机名、脚本名称和时间戳输出

您可以将 **Solaris Security Toolkit** 审计选项配置为包含主机名、脚本名称和详细级别 0 到 2 的时间戳信息。例如，如果要审计很多计算机，则您也许希望按照主机名、脚本名称或时间戳对输出进行排序。表 6-4 列出了这些变量。

表 6-4 显示主机名、脚本名称和时间戳审计输出

变量名	变量描述
JASS_DISPLAY_HOSTNAME	将此参数设置为 1 会使 Solaris Security Toolkit 软件将每个日志条目附加在系统的主机名之前。此信息基于 JASS_HOSTNAME 参数。默认情况下，该参数为空，因此 Toolkit 不会显示此信息。
JASS_DISPLAY_SCRIPTNAME	默认情况下，此参数值设置为 1，因此 Solaris Security Toolkit 软件会将每个日志附加在 audit 脚本名称之前。将该参数设定为其他任何值都会使 Toolkit 不显示此信息。
JASS_DISPLAY_TIMESTAMP	将该参数设置为 1 使 Solaris Security Toolkit 软件将每个日志附加在与审计运行操作相关的时间戳之前。此信息基于 JASS_TIMESTAMP 参数。默认情况下，该参数为空，因此该软件不会显示此信息。

通过配置 **Solaris Security Toolkit** 软件来附加主机、脚本和时间戳信息，可以将单个系统或者一组系统中的许多运行操作结合起来并按照关键数据对它们进行排序。可以使用此信息查找跨越几个系统的或者是在配置过程中有症状的问题。例如，以这种方式使用信息，管理员会能够辨别使用了给定进程的每个系统版本是否总是具有同样的故障检查。

通过将 JASS_DISPLAY_TIMESTAMP 参数设置为 1 以及将 JASS_DISPLAY_SCRIPTNAME 值设置为 0，会生成与以下内容类似的输出。

代码示例 6-5 审计日志条目的输出范例

```
# JASS_DISPLAY_TIMESTAMP=1
# JASS_DISPLAY_SCRIPTNAME=0
# export JASS_DISPLAY_TIMESTAMP JASS_DISPLAY_SCRIPTNAME
# bin/jass-execute -a abc.driver -V2
20050716132908 [FAIL] User adm is not listed in /etc/cron.d/at.deny.
20050716132908 [PASS] User bin is listed in /etc/cron.d/at.deny.
20050716132908 [FAIL] User zz999999 is not listed in /etc/cron.d/at.deny.
...
...
...
20050716132908 [FAIL] Script Total: 18 Errors
20050716132908 [FAIL] Driver Total: 28 Errors
20050716132908 [FAIL] Grand Total: 28 Errors
20050716132908 [SUMMARY] Results Summary for AUDIT run of dan.driver
20050716132908 [SUMMARY] The run completed with a total of 2 scripts run.
```

代码示例 6-5 审计日志条目的输出范例（续）

```
20050716132908 [SUMMARY] There were Failures in 2 Scripts
20050716132908 [SUMMARY] There were Errors in 0 Scripts
20050716132908 [SUMMARY] There were Warnings in 0 Scripts
20050716132908 [SUMMARY] There was a Note in 1 Script
20050716132908 [SUMMARY] Failure Scripts listed in:
/var/opt/SUNWjass/run/20050716132908/jass-script-failures.txt
20050716132908 [SUMMARY] Notes Scripts listed in:
/var/opt/SUNWjass/run/20050716132908/jass-script-notes.txt
#
```

进行安全性审计

在系统上定期进行安全性评估可提供一个基准，以判断安全性与所执行的安全性配置文件的匹配接近程度。安全性评估最通常的情形是在加强一个新安装的系统的安全性之后，作为一项安全性维护任务进行。安全评估选项可使您简化执行相同加强安全性驱动程序（用于加强系统安全性），而现在可使用 `-a` 选项与加强安全性过程中执行的安全性配置文件相比较以检查当前的状态。该设计消除了复杂性并提供了灵活性。例如，当升级安全性配置文件时，后续的安全性评估会使用更新的安全性配置文件。

在另外的可能情形中，可能应该对经过部署的系统安全性负责。在加强它们的安全性之前，需要进行安全性评估。在这种情形中，需要定义自己的安全性配置文件，定制一个 **Solaris Security Toolkit** 安全性配置文件模板或者按原样使用其中一个安全性配置文件模板。

▼ 进行安全性审计

在进行审计之前，需要定义或者选择一个安全性配置文件。有关详细信息，请参阅第 81 页“审计安全性的准备工作”。



注意 – 如果要已部署但尚未加强安全性的系统进行安全性评估，不需要先请加强安全性，请首先对计算机进行备份并重新引导以验证计算机处于一个已知、工作且一致的配置下。进行安全性评估之前，预备的重新引导过程中所遇的任何错误与警告都应得到更正并进行记录。

1. 选择要使用的安全性配置文件（加强安全性驱动程序）：

- 如果之前加强了系统安全，请使用同样的安全性配置文件。
例如，`secure.driver`。
- 如果您还没有加强系统安全性，请使用一个标准的安全性配置文件或者自己的文件。

例如, `secure.driver` 或者 `abccorp-secure.driver`。

对于标准的和产品特定的可用驱动程序, 有关它们完整的和最新的列表和信息, 请参阅 `security_drivers` 手册页或 《Solaris Security Toolkit 4.2 Reference Manual》中的第 4 章。

2. 确定所需的命令行选项和控制输出的方式。

请参见第 82 页 “使用选项和控制审计输出”。

3. 输入 `jass-execute -a` 命令、安全性配置文件名称和所需的选项。

以下是一个使用 `abc-secure.driver` 的审计运行操作范例。

代码示例 6-6 审计运行操作的输出范例

```
# ./jass-execute -a abc-secure.driver
[NOTE] Executing driver, abc-secure.driver

[...]

=====
abc-secure.driver: Audit script: enable-rfc1948.aud
=====

#-----
# RFC 1948 Sequence Number Generation
#
# Rationale for Verification Check:
#
# The purpose of this script is to verify that the system is
# configured and is in fact using RFC 1948 for its TCP sequence
# number generation algorithm (unique-per-connection ID). This is
# configured by setting the 'TCP_STRONG_ISS' parameter to '2' in
# the /etc/default/inetinit file.
#
# Determination of Compliance:
#
[...]
#-----

[PASS] TCP_STRONG_ISS is set to '2' in /etc/default/inetinit.
[PASS] System is running with tcp_strong_iss=2.

# The following is the vulnerability total for this audit script.

[PASS] Audit Check Total : 0 Error(s)

=====

# The following is the vulnerability total for this driver profile.
```


代码示例 6-6 审计运行操作的输出范例（续）

```
[PASS] Driver Total : 0 Error(s)

=====
abc-secure.driver: Driver finished.
=====

# The following is the vulnerability grand total for this run.

[PASS] Grand Total : 0 Error(s)
```

当审计运行操作启动后，**Solaris Security Toolkit** 软件从 `JASS_HOME_DIR/Audit` 目录中访问文件。虽然在 `JASS_HOME_DIR/Audit` 和 `JASS_HOME_DIR/Finish` 目录下的文件共享同样的基本文件名，但它们有不同的文件名后缀。按照指向文件（这些文件包含运行过程中在每个警报级别生成的所有消息）的指针，通过将后缀从 `.fin` 更改为 `.aud`，`driver.run` 脚本将 `JASS_SCRIPTS` 变量定义的 `finish` 脚本自动转换为 `audit` 脚本。

审计运行操作启动并初始化了 **Solaris Security Toolkit** 软件的状态。运行中所访问的每个驱动程序对所有文件模板与 `audit` 脚本的状态进行评估。每个检查会产生一个成功或者失败的结果，分别由攻击值 `0` 或者非 `0` 来代表。在大多数情况下，失败由数字 `1` 代表。每个运行脚本都根据脚本中包含的每次检查的攻击值的总和来产生总的安全性分数。由每个驱动程序引起的安全漏洞值总和会在每一个驱动程序评估完成时显示。运行结束时会显示重要的所有分数总和。

安全性评估选项提供了一个在评估运行操作启动后对系统状态的概述。**Solaris Security Toolkit** 软件通过检查配置文件来检查已存储的系统状态，并通过检查进程表信息、设备驱动程序信息等来检查正在运行的系统状态。**Solaris Security Toolkit** 软件检查每个文件及服务是否存在性，并且检查与服务相关的软件是否已经安装、配置、启用和运行。此方法为系统当前状态生成了一个准确的快照。

第7章

保护系统安全

本章介绍如何将前面章节提供的信息和专门技术应用到现实的环境中，以安装新的 Solaris 8 或 Solaris 9 OS 并保护其安全。本章举例说明如何用 Solaris 8 OS 的 Check PointFirewall-1 NG 来部署 Solaris Security Toolkit 软件。

本章的信息可用作保护新系统和应用程序安全的指南和案例。

Sun BluePrint 书籍和联机文章可为最小化系统和加强许多 Sun 系统的安全性提供指导。有关最新的产品专用的书籍和文章，请参阅以下 Web 站点：

<http://www.sun.com/blueprints>

本章包含以下主题：

- 第 93 页 “规划与准备”
- 第 95 页 “创建安全性配置文件”
- 第 95 页 “安装软件”
- 第 98 页 “配置 JumpStart 服务器和客户机”
- 第 103 页 “定制加强安全性配置”
- 第 107 页 “安装客户机”
- 第 108 页 “质量保证测试”

规划与准备

要按照该案例研究中所述有效地部署最小化的和安全的系统，规划和准备工作是很重要的。基本的网络体系结构、策略以及过程必须适当。另外，必须对系统的支持与维护进行定义与协调。有关规划与准备的更多信息，请参见第 2 章。本章中讲述的案例记录了系统管理员 (SA) 为防火墙系统的 Solaris OS 映像实现最小化系统和加强安全性所需执行的过程与任务。

在本案例中，系统管理员的任务是创建一种自动并可伸缩的解决方案，用于为服务提供商 (xSP) 构建和部署 Check PointFirewall-1 NG 系统，以便为其客户提供防火墙服务。对于本案例，xSP 的要求和注意事项如下：

- 因为 xSP 计划部署许多这种系统，因而创建与部署每个系统的时间是非常关键的，而且需要改进以提高效率。
- 需要使用与每个系统内部以太网的接口相连的专用管理网络来安装系统。网络仅可以由 xSP 员工使用，订户不能使用。
- 所有其他接口都位于独立的物理网络接口上，并且会被过滤掉。
- 网络管理的安全性对于经过部署的防火墙系统的整体安全性是至关重要的。

基于这些要求，系统管理员决定使用 **JumpStart** 技术与 **Solaris Security Toolkit** 软件以使 OS 映像的安装、最小化系统与加强安全性都实现自动化。

假设与限制

本章假设 **Solaris Security Toolkit** 软件已经运行并使用 **JumpStart** 技术进行安装。第 3 章提供了有关安装该软件的指导和规则。

本章还假设正在为最小化和加强某个特定应用程序的安全性而开发定制的配置。**Solaris Security Toolkit** 软件没有任何专门用于该程序的驱动程序或 **JumpStart** 配置文件。因此，需要为此应用程序创建定制的驱动程序和配置文件。要完成此任务，可复制现有的驱动程序和配置文件，然后针对该应用程序对它们进行修改。

此案例中，假设系统管理员 (SA) 具有以下熟练水平：

- 拥有足够的知识与经验进行 OS 与应用程序的配置。
- 了解如何对配置进行测试和微调。
- 了解如何从已经安装的客户机系统中构建一个 **JumpStart** 环境。请参阅 Sun BluePrints 书籍《**JumpStart Technology: Effective Use in the Solaris Operating Environment**》。
- 熟悉 OS 最小化技术。请参阅《**Enterprise Security: Solaris Operating Environment Security Journal, Solaris Operating Environment Versions 2.5.1, 2.6, 7, and 8**》。
- 熟悉 **Solaris Security Toolkit** 软件基础，且可随时使用最小化系统和加强安全性两种技术和准则来创建定制的配置。请参见第 1 章。

系统环境

案例实例基于以下硬件与软件环境：

- Check Point Firewall-1 NG
- Solaris 8 OS
- JumpStart 技术
- Solaris OS 群集 (SUNWCreg)
- Solaris Security Toolkit 软件
- 基于 SPARC 技术的平台

- 至少两个以太网接口

安全性要求

本案例中，已确定高级别要求和软件包，但是还需要确定所有软件包的特定组件和服务。另外，需要确定管理系统所需的 **Solaris OS** 功能。

以下列表提供了软件组件使用方法的详细信息：

- 用于远程管理的 **Secure Shell**
- 用于下载远程文件的 **FTP** 客户机
- 用于复制文件的 **scp** 和 **sftp** 命令

在此列表中，可以开发安全性配置文件。有关开发安全性配置文件和使用配置文件模板的详细信息，请参见第 26 页“开发和执行 **Solaris Security Toolkit** 配置文件”。

创建安全性配置文件

安全性配置文件定义了为加强和最小化系统安全性配置时 **Solaris Security Toolkit** 软件所做的安全性修改。任何包含在 **Solaris Security Toolkit** 软件中的标准安全性配置文件或者驱动程序都不能满足最小化 **Check Point Firewall-1 NG** 系统的要求。因而，必须创建一个定制的安全性配置文件以实现必需的系统修改。

对于本案例，本章中的几节介绍了创建安全性配置文件的流程。首先，创建基于现有驱动程序的新的驱动程序文件。然后修改新的驱动程序以满足之前所述的安全性要求。最小化系统在第 95 页“安装软件”中有讲述，而加强安全性修改在第 103 页“定制加强安全性配置”中有讲述。

安装软件

本节演示了安装软件的过程。对于范例案例，提供了关于异常情况或者特定案例的所有指导。有关安装软件的一般性指导，请参见第 3 章。

注 – 可将以下指导作为模板用来处理相关情况。

本节包含以下任务：

- 第 96 页“下载和安装安全性软件”

- 第 96 页 “安装修补程序”
- 第 97 页 “指定和安装 OS 群集”

下载和安装安全性软件

在 JumpStart 服务器上，下载并安装 Solaris Security Toolkit 及附加安全性软件（包括修补程序），如下所示。

▼ 下载和安装安全性软件

1. 下载 **Solaris Security Toolkit** 软件及附加安全性软件。
请参见第 35 页 “下载安全性软件”。
2. 安装已下载的 **Solaris Security Toolkit** 软件及附加安全性软件。
请参见第 42 页 “安装和执行软件”。



注意 – 目前先不要执行 Solaris Security Toolkit 软件。首先按照以下几节所述进行附加配置和定制。

安装修补程序

OS 修补程序可以解决安全性漏洞、可用性问题、性能问题或系统其他方面的问题。安装新的 OS 时，以及在 OS 安装后的运行过程中，都要进行检查以确保安装了所需的修补程序。

Solaris Security Toolkit 软件提供了一种机制，以安装联机的 SunSolve 上可用的推荐的修补程序群集和安全性修补程序群集。此 OS 专用的修补程序群集包含通常最需要的修补程序。

▼ 安装修补程序

1. 至少，将推荐的修补程序群集和安全性修补程序群集下载到 Patches 目录下并解压缩。
如果加强安全性驱动程序中包含了 `install-recommended-patches.fin` 脚本，则会自动安装修补程序群集。

对于 Check PointFirewall-1 NG，还存在一个问题。此应用程序需要推荐的修补程序群集和安全性修补程序群集中未包含的特定修补程序。Check PointFirewall-1 NG 需要以下修补程序：

- 108434
- 108435

2. 要自动安装修补程序 **108434** 和 **108435**，请从联机的 **SunSolve** 下载最新的版本并将它们放入 `Patches` 目录。

3. 用每个修补程序名称创建一个调用 `add_patch` 助手功能的全新 **finish** 脚本（例如，`fw1-patch-install.fin`）。

该 **finish** 脚本通过两个 Check PointFirewall-1 NG 所需修补程序 ID 号调用适当的助手功能。例如：

```
# !/bin/sh

# add_patch 108434-10

# add_patch 108435-10
```

指定和安装 OS 群集

为 OS 安装定义完磁盘布局后，下一个任务是指定需要安装哪一个 Solaris OS 群集。选择 Solaris OS 可用的五个安装群集之一：SUNWCreq、SUNWCuser、SUNWCprog、SUNWCall 和 SUNWCXall。

▼ 指定和安装 OS 群集

1. 指定要安装的 OS 群集。

由于本案例的目标是创建一个最小且专用的防火墙设备，因此请使用最小的可用 Solaris OS 群集，SUNWCreq，此软件包又称作 Core。

因为本群集包含了数量相对较少的软件包，所以也可能需要其他的软件包。这些所需的其他软件包必须包含在用 Solaris OS 群集定义的配置文件中。

基本的配置文件定义将以下内容添加到预先定义的配置文件中。

```
cluster          SUNWCreq
```

SUNWCreq 安装群集包含的软件包并非都是 Sun 防火墙服务器正常运行所需要的。在具有了工作基准后，请删除这些额外的软件包。请参阅 Sun BluePrints OnLine 文章 "Minimizing the Solaris Operating Environment for Security: Updated for the Solaris 9 Operating Environment"。

2. 用定义好的安全性配置文件进行安装以确定是否存在软件包依赖性问题。

如果在安装过程中遇到一些软件包依赖性问题，则认为 Check PointFirewall-1 NG 需要以下 Solaris OS 软件包：

- SUNWter — 终端信息
- SUNWadm — 系统管理核心库
- SUNWadmfw — 系统和网络管理框架
- SUNWlibC 和 SUNWlibCx — Check PointNG 应用程序程序所需软件包

配置文件中的软件包完整列表如下。

cluster	SUNWCreq	
package	SUNWter	add
package	SUNWlibC	add
package	SUNWlibCx	add
package	SUNWadm	add
package	SUNWadmfw	add

注 — 虽然对于本案例研究而言，此列表是完整的，然而要根据部署此配置的实际环境来添加或删除附加的软件包。

软件包的最终列表可能还需要修改，直到按第 108 页“质量保证测试”中所述从功能及安全性两方面对系统进行了验证。若如此，请修改配置文件，重新安装系统然后重复该测试。

3. 基于之前两个步骤中的软件包依赖性，创建一个 minimize-firewall.fin 脚本。

配置 JumpStart 服务器和客户机

本节演示了如何配置 JumpStart 服务器和客户机，以使用一个定制的安全性配置文件进行最小化。有关在 JumpStart 环境下使用 Solaris Security Toolkit 软件的详细信息，请参见第 5 章。

本节包含以下任务：

- 第 99 页“准备基础结构”
- 第 101 页“验证和检查 rules 文件”

准备基础结构

执行以下任务以准备基础结构。以下任务演示了为在客户机上使用现有驱动程序、配置文件和 **finish** 脚本而创建基准配置的过程。确定基准后，验证其运转是否正常，然后为所选的应用程序定制基准。

▼ 准备基础结构

1. 配置 JumpStart 服务器和环境。

有关详细的操作指导，请参阅第 5 章。

2. 使用 `add-client` 命令向 JumpStart 服务器中添加客户机。

代码示例 7-1 在 JumpStart 服务器上添加一台客户机

```
# pwd
/jumpstart
# bin/add-client -c jordan -o Solaris_8_2002-02 -m sun4u
-s nomex-jumpstart
cleaning up preexisting install client "jordan"
removing jordan from bootparams
updating /etc/bootparams
```

3. 为客户机创建一个 `rules` 文件条目，指定适当的 JumpStart 配置文件和 **finish** 脚本。例如：

```
hostname jordan - Profiles/xsp-minimal-firewall.profile \
Drivers/xsp-firewall-secure.driver
```

4. 通过复制 Solaris Security Toolkit 软件提供的文件，创建名为 `xsp-minimal-firewall.profile` 的配置文件和名为 `xsp-firewall-secure.driver` 的驱动程序文件。

在成功完成下一步之前必须创建这些文件。最初，这些文件可能是与 Solaris Security Toolkit 软件一起发行的文件的副本。

注 – 请勿修改与 Solaris Security Toolkit 软件一起发行的原始文件。

以下实例显示了如何创建文件。

代码示例 7-2 创建一个配置文件

```
# pwd
/jumpstart/Drivers
# cp install-Sun_ONE-WS.driver xsp-firewall-secure.driver
# cp hardening.driver xsp-firewall-hardening.driver
[...]
# pwd
/jumpstart/Profiles
# cp minimal-Sun_ONE-WS-Solaris8-64bit.profile \
    xsp-minimal-firewall.profile
```

本实例以一个专用的 Web 服务器配置为基础，因为它是开发专用防火墙的很好的基准。

5. 创建配置文件和驱动程序文件后，请按照以下所示修改文件：

- a. 用 `xsp-firewall-hardening.driver` 替换与 `hardening.driver` 相关的 `xsp-firewall-secure.driver`。
- b. 替换定义在 `JASS_SCRIPTS` 中的与 `minimize-firewall.fin` 和您的 **finish** 脚本（例如，`fwl-patch-install.fin`）相关的两个 **finish** 脚本。

修改的脚本应与以下显示相似：

代码示例 7-3 修改的脚本的输出范例

```
DIR="`/bin/dirname $0`"
export DIR
. ${DIR}/driver.init
. ${DIR}/config.driver
JASS_SCRIPTS="
                minimize-firewall.fin
                fwl-patch-install.fin"
. ${DIR}/driver.run
. ${DIR}/xsp-firewall-hardening.driver
```

6. 使用以下命令检查 rules 文件条目以纠正错误。

代码示例 7-4 检查 rules 文件以纠正错误

```
# pwd
/jumpstart
# ./check
Validating rules...
Validating profile Profiles/end-user.profile...
Validating profile Profiles/xsp-minimal-firewall.profile...
Validating profile Profiles/test.profile...
Validating profile Profiles/entire-distribution.profile...
Validating profile Profiles/oem.profile...
The custom JumpStart configuration is ok.
```

这时，应该能够在客户机（本例中为 jordan）上开始 JumpStart 安装。使用 JumpStart 配置和创建的 Solaris Security Toolkit 驱动程序、finish 脚本与配置文件。

7. 如果在检查 rules 文件时遇到问题，请参见第 101 页“验证和检查 rules 文件”。
8. 在客户机的 ok 提示符下，输入以下命令以使用 JumpStart 基础结构安装客户机。

```
ok> boot net - install
```

如果不能构建客户机，请检查配置并对其进行修改直到它能正常运行。注意，本节并未提到 JumpStart 配置的所有方面。有关更多细节，请参阅 Sun BluePrints 书籍《JumpStart Technology: Effective Use in the Solaris Operating Environment》。

正确运行 rules 文件并验证所安装的修补程序是正确的之后，可以开始进行客户机系统基础级别的安装并最小化系统和加强安全性。

验证和检查 rules 文件

当验证 rules 文件以纠正错误时，可能会遇到多种问题。本节提到了一些最常见的问题。

第一次运行 `rules` 文件会产生以下输出结果。

代码示例 7-5 `rules` 文件的输出范例

```
# pwd
/jumpstart
# ./check
Validating rules...
Validating profile Profiles/xsp-minimal-firewall.profile...
Error in file "rules", line 20
hostname jordan - Profiles/xsp-minimal-firewall.profile
Drivers/xsp-firewall-secure.driver
ERROR: Profile missing:
    Profiles/xsp-minimal-firewall.profile
```

在本例中，不存在 `jordan` 的 `rules` 条目中指定的配置文件。配置文件 `xsp-minimal-firewall.profile` 没有在 `Profiles` 目录中出现。典型地，此错误产生的原因是文件名拼写错误，忘记为配置文件指定正确的路径或者还没有创建配置文件。修正错误并再次进行检查。

第二次运行操作会暴露两个其他问题。第一个问题是在 `xsp-firewall-secure.driver` 中调用的驱动程序。`xsp-firewall-secure.driver` 没有调用 `xsp-firewall-hardening.driver`，而仍然调用 `hardening.driver`。

第二个问题是将 `JASS_SCRIPTS` 变量错误地设置为 `minimize-Sun_ONE-WS.fin` 而不是 `minimize-firewall.fin`。

以下是错误的脚本。

代码示例 7-6 错误脚本范例

```
#!/bin/sh
DIR="/bin/dirname $0`"
export DIR
. ${DIR}/driver.init
. ${DIR}/config.driver
JASS_SCRIPTS="minimize-Sun_ONE-WS.fin"
. ${DIR}/driver.run
. ${DIR}/hardening.driver
```

以下是一个正确脚本的实例。

代码示例 7-7 正确脚本范例

```
#!/bin/sh
DIR="/bin/dirname $0"
export DIR
. ${DIR}/driver.init
. ${DIR}/config.driver
JASS_SCRIPTS="
minimize-firewall.fin"
. ${DIR}/driver.run
. ${DIR}/xsp-firewall-hardening.driver
```

定制加强安全性配置

所建议防火墙的加强安全性配置可随时进行定制与调整。初始脚本基于 `hardening.driver`。这意味着系统进入了“屏蔽墙”，也就是说系统的所有服务都被禁用。

由于 Solaris 8 OS 不包含 Secure Shell 客户机，因此需要进行修改以允许对防火墙进行远程的基于网络的管理。本案例中提到的防火墙，要求 FTP 服务必须保持启用而且必须安装 Secure Shell 客户机以进行远程管理。将这两项服务限制到仅用于私用管理网络，从而不启动对任何其他网络接口的监听。有关限制这些服务的信息，请参阅 Sun BluePrints OnLine 中标题为 "Solaris Operating Environment Security: Updated for Solaris 9 Operating Environment" 的文章。

除了将这两项服务置于启用状态外，也将 RPC 服务置于启用状态，以便能够使用 Solstice DiskSuite GUI 将 Solstice DiskSuite 配置为进行磁盘镜像配置。如果不准备使用 Solstice DiskSuite GUI，则不需要 RPC 服务。本例中，需要使用 GUI 因而要保持启用 RPC 服务。请注意，Solstice DiskSuite 的安装与配置超出了本书的范围。

本客户机所需的最终修改是使用 xSP 的中央 SYSLOG 服务器手动定制 `syslog.conf` 脚本。该定义的 `syslog.conf` 文件必须安装在防火墙系统中的每台计算机上。

这些修改需要更改许多 Solaris Security Toolkit 配置选项。在以下几节中详细介绍每个所需的修改。

- 第 104 页 “启用 FTP 服务”
- 第 104 页 “安装 Secure Shell 软件”
- 第 105 页 “启用 RPC 服务”
- 第 106 页 “定制 `syslog.conf` 文件”

启用 FTP 服务

对于本案例中的防火墙，请保留 FTP 服务处于启用状态。

▼ 启用 FTP 服务

1. 要保留 FTP 服务处于启用状态，请设置 `JASS_SVCS_DISABLE` 和 `JASS_SVCS_ENABLE` 变量以修改 `update-inetd-conf.fin` 文件的默认操作。
要禁用除 FTP 之外所有标准的 Solaris OS 服务，对于本案例的最佳方法是将 `JASS_SVCS_ENABLE` 定义为 `ftp`，而确保 `JASS_SVCS_DISABLE` 保留为从 `finish.init` 脚本中获得的默认值。请参阅《Solaris Security Toolkit 4.2 Reference Manual》。
2. 要通过环境变量实现更改，请在调用 `xsp-firewall-hardening.driver` 之前，向 `xsp-firewall-secure.driver` 添加一条与以下相似的条目。

```
JASS_SVCS_ENABLE="ftp"
```

3. 通过防火墙软件实现 FTP 服务以确保它仅在 xSP 的管理网络上可用。

其中的另一项要求是 FTP 应该仅在 xSP 的管理网络上可用。在 Solaris 8 OS 中，既可以通过在系统上使用 TCP 封装器，又可以通过防火墙软件本身来实现此要求。在本案例中，是通过防火墙软件实现此要求的。

安装 Secure Shell 软件

注 – 这些指导仅适用于运行 Solaris 8 OS 的系统。如果系统当前运行的是 Solaris 9 或 10 OS，则可以使用 Solaris OS 附带的 Secure Shell 软件并跳过本节中的 OpenSSH 安装。

Solaris 8 OS 不包含 Secure Shell 客户机，因此如果您的系统运行的是 Solaris 8 OS，则需要安装 Secure Shell 客户机以便进行远程管理。

可以配置 Solaris Security Toolkit 软件以安装 OpenSSH 工具。请使用列在 `config.driver` 文件（由 `xsp-firewall-secure.driver` 使用）中的 `install-openssh.fin` 脚本。

▼ 安装 Secure Shell

1. 将默认的 `config.driver` 复制到 `xsp-firewall-config.driver`。
2. 在文件的副本中，去掉 `install-openssh.fin` 的注释条目。
3. 修改 `xsp-firewall-secure.driver` 的条目使其原本调用 `config.driver` 转而调用 `xsp-firewall-config.driver`。

4. 获得 **OpenSSH** 的最新版本。

与修补程序和 OS 版本一样，请使用 **OpenSSH** 的最新版本。有关最新版本信息，请参阅 **OpenSSH** 网页：

<http://www.openssh.org>

5. 编译最新的 **OpenSSH** 软件包，进行命名并将它安装到 `Packages` 目录下。

有关本软件包的更多信息，请参阅 **Sun BluePrints OnLine** 中标题为 "Configuring **OpenSSH** for the Solaris Operating Environment" 的文章。

6. 更新 `install-openssh.fin` 脚本以反映正确的 **OpenSSH** 软件包名称。

可能需要修改 `install-openssh.fin` 脚本。该脚本将 **OpenSSH** 软件包名称定义为与以下格式相似：

```
OBSDssh-3.5p1-sparc-sun4u-5.8.pkg
```

其中，软件包名称后紧跟版本号 (3.5p1)、体系结构 (sparc)、体系结构版本号 (sun4u)、软件包编译所针对的操作系统 (5.8) 和一个 `pkg` 后缀。

7. 通过防火墙软件实现 **SSH** 以确保它仅在 **xSP** 的管理网络上可用。

其中声明的另一项要求是 **Secure Shell** 仅在 **xSP** 的管理网络上可用。在 **Solaris 8 OS** 中，既可以通过系统使用 **TCP** 封装器，又可以通过防火墙软件本身来实现此要求。在本案例中，是通过防火墙软件实现此要求的。请注意也可以通过修改 **Secure Shell** 服务器的配置来实现此项要求。

启用 RPC 服务

将 **RPC** 服务置于启用状态，从而可以使用 **Solstice DiskSuite** 做磁盘镜像，这需要 **RPC** 服务。

因为在 **Solaris Security Toolkit** 运行过程中可使用一个特定 `finish` 脚本 `disable-rpc.fin` 禁用 **RPC** 服务，所以此修改是相对简单的。

注 - 系统的防火墙配置应明确拒绝远程访问系统 **RPC** 服务。

▼ 启用 RPC

- 为 `xsp-firewall-hardening.driver` 中的 `disable-rpc.fin` 的条目加注释。

通过为驱动程序中的脚本加注释符而不是直接删除它们来最终禁用这些脚本。在为 `JASS_SCRIPTS` 中定义的条目添加注释符时要小心，因为它只接受特定的注释值的组合。

以下为包含在 `driver.funcs script` 中的注释，说明了 **Solaris Security Toolkit** 软件可作为注释而接受的定义在 `JASS_SCRIPTS` 中的注释符。

```
#Very rudimentary comment handler. This code will only recognize
#comments where a single '#' is placed before the file name
#(separated by white space or not). It then will only skip the
#very next argument.
```

定制 `syslog.conf` 文件

本客户机所需的最终修改是使用 **xSP** 的中央 `SYSLOG` 服务器手动定制的 `syslog.conf` 脚本。该定义的 `syslog.conf` 文件必须安装在防火墙系统中的每台计算机上。

▼ 定制 `syslog.conf` 文件

1. 复制 **xSP** 的标准 `syslog.conf` 文件，将它重命名为 `syslog.conf.jordan`，再将其放到 `Files/etc` 目录中。

Solaris Security Toolkit 软件支持几种不同的复制文件模式。对于此配置最适当的选项是将系统主机名作为后缀附加到文件，以便只将 `syslog.conf` 文件复制到 `jordan`，因为它有唯一的防火墙专用修改。在本例中，客户机叫作 `jordan`，因而 `Files/etc` 中使用的实际文件名是 `syslog.conf.jordan`。注意，请勿为 `JASS_FILES` 定义添加此后缀，这点是非常重要的。有关后缀的更多信息，请参阅《**Solaris Security Toolkit 4.2 Reference Manual**》。

2. 如果 **xSP** 标准的 `syslog.conf` 文件不可用，则按照如下所示创建一个定制文件 `syslog.conf`。
 - a. 复制包含在 **Solaris Security Toolkit** 中的 `syslog.conf` 文件，然后将它重新命名为 `syslog.conf.jordan`，再将其放到 `Files/etc` 目录中。
 - b. 修改 `syslog.conf.jordan` 以使它符合 **xSP** 有关 `SYSLOG` 的标准。

3. 验证 `/etc/syslog.conf` 文件已列在 `xsp-firewall-hardening.driver` 的 `JASS_FILES` 定义中。

默认情况下，`xsp-firewall-hardening.driver` 中的 `JASS_FILE` 定义会显示为如下。

代码示例 7-8 修改的 `xsp-firewall-hardening.driver` 输出范例

```
JASS_FILES="
                /etc/dt/config/Xaccess
                /etc/init.d/inetsvc
                /etc/init.d/nddconfig
                /etc/init.d/set-tmp-permissions
                /etc/issue
                /etc/motd
                /etc/notrouter
                /etc/rc2.d/S00set-tmp-permissions
                /etc/rc2.d/S07set-tmp-permissions
                /etc/rc2.d/S70nddconfig
                /etc/syslog.conf
"
```

这时，所有需要的修改已完成。OS 的安装、最小化和加强安全性操作已为特定的应用程序进行了定制并完全自动化。唯一没有完全自动化的过程是防火墙软件和 **Solstice DiskSuite** 的配置及安装。虽然可以使用 **JumpStart** 技术进行这些配置，但已超出了本书的范围。请参阅 **Sun BluePrints** 书籍《**JumpStart Technology: Effective Use in the Solaris Operating Environment**》。

安装客户机

对驱动程序做完所有修改后，请按本节所述安装客户机。

▼ 安装客户机

1. 对驱动程序进行所有要求的修改后，请使用 **JumpStart** 基础结构安装客户机。在客户机中的 `ok` 提示符下使用以下命令。

```
ok> boot net - install
```

2. 如果出现错误，请修正它们并重新安装客户机的 OS。

质量保证测试

保护系统安全过程的最终任务涉及验证系统提供的应用程序和服务是否正常运行。同时，此任务还验证安全性配置文件是否成功地实现了所需的修改。

非常重要的一点是，重新引导已加强安全性并最小化的平台后要全面完成此任务，以确保可以检测到并能够迅速纠正出现的任何异常和问题。本过程分为两个任务：验证配置文件安装和验证应用程序与服务的功能。

▼ 验证配置文件安装

要验证 **Solaris Security Toolkit** 软件是否正确且没有任何错误地安装了安全性配置文件，请按照以下步骤进行检查与评估。

1. 检查安装日志文件。

此文件安装在 `JASS_REPOSITORY/jass-install-log.txt` 中。

注 – 此日志文件可以作为参考以准确理解 **Solaris Security Toolkit** 软件对系统所做的工作。对于在系统上运行的每个操作，都会根据该运行操作的起始时间在目录中保存一个新的日志文件。**请勿**直接修改这些文件和 `JASS_REPOSITORY` 目录中的其他文件。

2. 使用审计选项评估系统的安全性配置。

有关审计选项的详细信息，请参见第 6 章。本案例中，使用安装了 **Solaris Security Toolkit** 软件的客户机目录中的以下命令。

代码示例 7-9 评估安全性配置

```
# ./jass-execute -a xsp-firewall-secure.driver
[NOTE] Executing driver, xsp-firewall-secure.driver
=====
xsp-firewall-secure.driver: Driver started.
=====

=====
Solaris Security Toolkit Version:   4.2.0
[...]
```

如果在对 **Solaris Security Toolkit** 运行操作验证时遇到任何不一致，请记下它们。运行操作结尾处的摘要会报告所发现的不一致总数。运行操作的全部输出位于 `JASS_REPOSITORY` 目录中。

▼ 验证应用程序和服务的功能

应用程序和服务的验证过程包括一个明确的测试和一个验收规划。此规划用于使用系统中各种各样的组件或者应用程序以确定它们可用且运行良好。如果这样的规划不可用，则要根据系统的使用方式采取合理的方法对系统进行测试。目的是确保加强安全性过程决不会影响应用程序或服务执行其功能。

1. 如果在加强系统安全性之后，发现了应用程序或者服务的故障，请使用第 2 章中所述的技术确定问题。

例如，使用 `truss` 命令。此命令通常用于确定应用程序出现问题的故障点。一旦了解后，即可找到问题并跟踪返回到 **Solaris Security Toolkit** 软件所做的更改。

注 – 根据许多部署过 **Solaris Security Toolkit** 软件人员的共同经验，使用本书中的方法可以避免大多数问题。

2. 以类似的方式测试 **Check Point Firewall-1 NG** 软件，并将跟踪问题返回到 **Solaris Security Toolkit** 软件所做的修改然后纠正这些问题。
3. 如果软件包的最终列表需要修改，则修改配置文件，重新安装系统再重复测试。

词汇表

本列表定义了 Solaris Security Toolkit 中涉及的单词缩写和首字母缩写词。

A

- ab2 AnswerBook2
- ABI** Application Binary Interface (应用程序二进制接口)
- ARP** Address Resolution Protocol (地址解析协议)
- ASPPP** Asynchronous Point-to-Point Protocol (异步点对点协议)

B

- BART** Basic Auditing and Reporting Tool (基本审计和报告工具)
- BIND** Berkeley Internet Name Domain (Berkeley Internet 名称域)
- BSD** Berkeley Software Distribution (Berkeley 软件分发)
- BSM** Basic Security Model (基本安全模型) (*Solaris*)

C

- CD** compact disc (紧凑型光盘)

CD-ROM compact disc-read-only memory (紧凑型光盘—只读存储器)
CDE Common Desktop Environment (公用桌面环境)
cp(1) 复制文件命令
cron(1M) 时钟守护进程命令

D

DHCP Dynamic Host Configuration Protocol (动态主机配置协议)
DMI Desktop Management Interface (桌面管理界面)
DMTF Distributed Management Task Force (分布式管理任务组)
DNS Domain Name System (域名系统)

E

EEPROM electronically erasable programmable read-only memory (加电可擦写的可编程只读存储器)

F

FACE Framed Access Command Environment (框架访问命令环境)
FMRI Fault Management Resource Identifier (故障管理资源标识符)
FTP File Transfer Protocol (文件传输协议)

G

GID group identifier (组标识)
GUI graphical user interface (图形用户界面)

H

- HSFS** High Sierra File System (High Sierra 文件系统)
- HTTP** HyperText Transfer Protocol (超文本传输协议)

I

- ID** identifier (标识符)
- IETF** Internet Engineering Task Force (Internet 工程任务组)
- INETD** Internet service daemon (Internet 服务守护进程)
- IP** Internet Protocol (Internet 协议)
- IPF** Internet Protocol Filter (Internet 协议过滤器)
- ISA** instruction set architecture (指令集体系结构)

J

- JASS** JumpStart Architecture and Security Scripts (JumpStart 体系结构和安全脚本), 现在是 Solaris Security Toolkit

K

- KDC** Kerberos Key Distribution (Kerberos 密钥分发)

L

- LDAP** Lightweight Directory Access Protocol (轻量目录访问协议)
- lp(1)** 行式打印机命令 (提交打印请求)

M

- MAN** 管理网络 (*Sun Fire* 高端系统内部 *I1* 网络)
- MD5** message-digest 5 algorithm (message-digest 5 算法)
- MIP** Mobile Internet Protocol (移动 Internet 协议)
- MSP** midframe service processor (中型服务处理器)
- mv(1)** 移动文件命令

N

- NFS** Network File System (网络文件系统)
- NG** Next Generation (下一代)
- NGZ** non-global zone (非全局区域)
- NIS, NIS+** Network Information Services (网络信息服务)
- NP** no password (无密码)
- NSCD** name service cache daemon (名称服务高速缓存守护进程)

O

- OEM** Original Equipment Manufacturer (原始设备制造商)
- OS** Operating System (操作系统)

P

- PAM** Pluggable Authentication Module (可插拔验证模块)
- PDF** Portable Document Format (可移植文档格式)
- Perl** Practical Extraction and Report Language (实用提取和报告语言)

- PICL** Platform Information and Control Library (平台信息和控制库)
PPP Point-to-Point Protocol (点对点协议)
PROM Point-to-Point Protocol (可编程只读存储器)
-

Q

- QA** quality assurance (质量保证)
-

R

- RBAC** role-based access control (基于角色的存取控制)
rc 运行控制 (文件或脚本)
rlogin(1) 远程登录命令
RFC Remote Function Call (远程功能调用)
RPC Remote Procedure Call (远程过程调用)
rsh(1) 远程 shell 命令
-

S

- SA** system administrator (系统管理员)
SC 系统控制器 (*Sun Fire* 高端系统和中型系统)
scp(1) 安全复制命令 (远程文件复制程序)
SCCS Source Code Control System (源代码控制系统)
SLP Service Location Protocol (服务定位协议)
SMA System Management Agent (系统管理代理)
SMC Solaris Management Console (*Solaris* 管理控制台)
SMF Service Management Facility (服务管理工具)
SMS System Management Services (系统管理服务)

-
- SNMP** Simple Network Management Protocol (简单网络管理协议)
 - SP** service provider (服务提供商)
 - SPARC** Scalable Processor Architecture (可伸缩的处理器体系结构)
 - SPC** SunSoft Print Client (SunSoft 打印客户机)
 - SSH** 安全 Shell (*Solaris 9* 和 *10 OS*)
 - SVM** Solaris Volume Manager (Solaris 卷管理器)

T

- TCP** Transmission Control Protocol (传输控制协议)
- tftp(1)** trivial file transfer program (琐碎文件传输程序)
- ttl** time-to-live (存在时间)

U

- UDP** User Datagram Protocol (用户图表协议)
- UFS** Unix File System (Unix 文件系统)
- UID** user identifier (用户标识)
- UUCP** UNIX-to-UNIX Copy (UNIX 对 UNIX 复制)

V

- VOLD** Volume Management daemon (Volume Management 守护进程)

W

- WBEM** Web-based Enterprise Management (基于 Web 的企业管理)

索引

符号

/usr/bin/ldd 命令, 19

A

add_install_client 命令, 76

add_to_manifest 功能, 59

add-client 脚本, 3, 76

案例, 93

安全, 维护, 29, 79

安全策略

标准, 15

查看, 17

开发, 17

安全漏洞

策略, 29

分析, 16

扫描, 16

值, 定义的, 91

安全配置, 评估, 28

安全性

要求, 15

安全性配置文件

创建, 案例, 95

模板, 81

默认, 29

嵌套或分层, 26

验证, 55

验证安装, 案例, 108

安全性评估

配置, 54

执行, 89

安全性软件, 下载, 35

安全性状态

查看, 80

审计, 80

安全性, 监视, 29

安装

安装前任务, 27

备份, 27

标准化, 71

规划, 31

加强系统, 32

客户机, 案例, 107

日志文件, 28

软件, 26

软件, 案例, 95

使 Solaris OS 自动化, 11

新系统, 案例, 93

修补程序, 10

验证, 27

之后进行审计, 89

准则, 2

自动化, 2, 71

自动化修补程序, 10

安装前任务, 27

B

- b 选项, 撤消, 61
- backup_file 助手功能, 59
- BSM, 39
- 版本控制, 11
- 报告, 电子邮件通知, 62
- 保护经过部署的系统的的功能, 16
- 保护系统安全, 方法论, 15
- 保留选项, 62
- 备份
 - 安装前, 27
 - 撤消运行前的要求, 63
 - 审计, 89
- 备份软件, 清点, 18
- 备份文件
 - 默认操作, 58
- 被利用的系统, 16
- 编译器, 安装警告, 40
- 编译器, 限制, 40
- 标准, 安全策略, 17
- 标准, 跨平台强制执行, 26
- 不能构建客户机, 案例, 101
- 部署系统, 71
- 部署最小化和安全的系统, 93
- 不一致的状态, 62

C

- Check Point Firewall-1 NG, 93
- clients
 - 从 JumpStart 服务器删除, 77
 - 从 JumpStart 服务器添加, 76
- core.profile, 74
- cp 命令, 59
- cron 作业
 - 审计运行, 80
 - 使用抑制显示输出选项, 62
- 测试功能, 28
- 测试和验收计划, 28
- 测试, 在非生产系统上, 37

- 查看日志文件, 28
- 差异, 发现, 54
- 超时, 程序, 23
- 撤消
 - 保留选项, 62
 - 备份选项, 61
 - 不可用, 58
 - 撤消运行, 63
 - 电子邮件选项, 62
 - 记录并恢复更改, 57
 - 交互式运行, 60
 - 命令行选项, 52
 - 强制选项, 61
 - 使用所需信息, 57
 - 手动撤消更改, 59
 - 输出选项, 62
 - 限制, 58
 - 选项, 61
 - 选择运行, 输出范例, 64
 - 抑制显示输出选项, 62
 - 运行, 列表, 64
- 创建安全性配置文件, 案例, 95
- 存储的状态, 91
- 错误
 - 解析 sysidcfg 文件时, JumpStart 模式, 73
 - 破坏内容, 57, 58
 - 系统崩, 57
 - 系统崩溃, 58
 - 消息或警告, 27
- 错误修正, 修补程序, 37
- 重新引导, 保护系统安全, 16

D

- d 驱动程序选项限制, 49
- Developer Solaris OS 群集, SUNWCprog, 75
- developer.profile, 75
- DNS 服务, 22
- documentation 目录, 4
- driver.init 文件
 - 概述, 5
- dtexec 进程, 25

- 电子邮件通知选项, 50
- 调试服务, 24
- 定期审计, 80
- 定制
 - 安全性审计, 80
 - 策略和要求, 12
 - Solaris Security Toolkit, 12
 - syslog.conf 文件, 106
 - 准则, 12
- 独立模式, 32
 - 使用, 44
 - 执行, 45
- 端口, 确定使用情况, 25
- 对审计输出进行排序, 88
- 对使用的审计, 15
- 多宿主的 JumpStart 服务器, 73

E

- End User Solaris OS 群集, SUNWCuser, 74
- end-user.profile, 74
- Entire Distribution Solaris OS 群集, SUNWCall, 75
- entire-distribution.profile, 75
- 二进制, 验证, 41

F

- f 选项, 撤消, 61
- files 目录, 8
- finish 脚本
 - 撤消功能, 59
 - 新建, 58
- Finish 目录, 9
- finish.init 文件
 - 驱动程序流程, 5
- FixModes
 - FixModes.tar.Z 文件, 38
 - 软件, 下载, 38
- FTP
 - 服务, 启用, 案例, 104
 - 默认配置, 17

- 返回值, 21
- 范例, 配置文件, 74
- 方法论, 保护系统安全, 15
- 访问权限, 保护, 38
- 访问网络, 保护, 39
- 风险和益处, 考虑, 15
- 服务
 - 标识, 16
 - 根据需要确定, 24
 - 清点, 18
 - RPC, 103
 - 退出, 挂起, 或退出, 22
 - 限制, 103
 - 要求, 16
 - 最近使用的, 确定, 24
- 服务框架, 22
- 服务要求, 确定, 18

G

- 跟踪更改, 57
- 更改控制策略, 27
- 更改原始文件, 13
- 更强的身份验证, 17
- 工具, 可选的, 41
- 功能性
 - 测试, 28
 - 添加, 81
 - 问题, 16
 - 修补程序, 37
- 共享库, 19
- 故障, 29
- 故障, 应用程序, 28
- 故障检查, 88
- 故障排除, 16
 - 撤消运行, 60
 - 系统修改, 54
- 关键环境变量, 26
- 关键组件, 1
- 管理软件, 清点, 18
- 管理协议, 策略实例, 17

规划阶段, 15
规划与准备, 案例, 93
规划, 安装, 31

H

后门访问, 二进制文件, 40
环境变量
 导入, 7
环境, 配置, 32
恢复更改, 57

J

JASS, 1

JASS_DISPLAY_HOSTNAME 变量, 88
JASS_DISPLAY_SCRIPTNAME 变量, 88
JASS_DISPLAY_TIMESTAMP 变量, 88
JASS_HOME_DIR 环境变量, 定义, 36
JASS_LOG_BANNER 环境变量, 85, 86
JASS_LOG_ERROR 环境变量, 86
JASS_LOG_FAILURE 环境变量, 86
JASS_LOG_SUCCESS 环境变量, 86
JASS_LOG_WARNING 环境变量, 86
JASS_REPOSITORY
 撤消运行, 57
 检查内容, 60
 修改内容, 57
jass-check-sum 程序, 3
jass-check-sum 命令, 59
jass-execute -a 命令, 90
jass-execute -a 命令选项, 82
jass-execute 命令选项, 44
jass-execute -u 命令, 60
jass-manifest.txt 文件, 57
jass-undo-log.txt 文件, 64
基本安全模块 (BSM), 39
基础结构, 15
基础结构组件, 18
基础结构, 准备, 案例, 99

记录结果, 22

JumpStart 服务器

多宿主的, 73
配置与管理, 71
配置, 案例, 98

JumpStart 技术, 33, 71

JumpStart 技术, 受支持的 OS 版本, 71

JumpStart 客户机

安装客户机, 案例, 107
不能构建, 案例, 101
添加, 案例, 99
文件, 存储, 8

JumpStart 模式

安装, sysidcfg 目录, 11
解析 sysidcfg 文件时出现错误, 73
配置, 33, 72
使用所选脚本, 73
使用所有脚本, 73
修改 sysidcfg, 72

JumpStart 配置文件, 74

模板, 74
目录, 10

JumpStart 体系结构和安全脚本 (JASS), 1

JumpStart 体系结构, 集成 Solaris Security Toolkit,
72

基于主机的访问控制, 16

集中的 syslog 系统信息库, 29

加密, 17

加密软件, 39

加强运行

 撤消列表, 64

 恢复更改, 63

 执行 Solaris Security Toolkit, 42

假设与限制, 案例, 94

鉴别

 服务, 22

 更强, 17

 强, 39

检查

 failed, 88

 添加, 81

检查安全性状态, 80

- 监视安全, 29
- 监视软件, 清点, 18
- 脚本
 - 列表, 5
 - 命名, 13
 - 修改, 警告, 73
- 较慢的网络连接, 使用抑制显示输出, 62
- 结构, 软件, 3
- 结果, 记录, 22
- 进程
 - 标识符, 21
 - 确定哪些正在使用文件和端口, 25
- 警告消息
 - 在系统引导或应用程序启动时显示, 27
 - 执行 Solaris Security Toolkit 软件, 38
- 经过部署的系统
 - 安装软件, 26
 - 保护安全, 16

K

- k 选项, 撤消, 62
- Kerberos, 17
- kill 命令, 23
- 可操作的或管理功能, 清点, 18
- 库, 共享, 19
- 快速加强系统, 33
- 框架, 服务, 22
- 框架, 自定义 Solaris Security Toolkit, 59
- 扩展, 17

L

- LDAP, 22
- ldd 命令, 24
- librpcsvc.so.1 条目, 24
- lsnf 程序, 25
- lsnf 程序, 获得, 25
- 历史记录选项, 50
- 列出打开的文件的程序, 25

M

- m 选项
 - 撤消, 62
 - 审计, 83
- make-jass-pkg 程序, 3
- man 目录, 5
- MD5 二进制文件, 40
- MD5 软件
 - md5.tar.z 文件, 41
 - 下载, 40
- 密码
 - 策略实例, 17
 - passwd(1) 命令, 18
- 命令行选项
 - 撤消, 52, 61
 - 电子邮件通知, 50
 - help, 48
 - jass-execute 命令, 43
 - 历史记录, 50
 - 驱动程序, 49
 - root, 52
 - 审计, 46, 82
 - 审计, 帮助, 82
 - 输出文件, 51
 - 抑制显示输出, 51
 - 最近执行, 50
- 命名标准
 - 安装, 9
 - 定制文件, 13
 - Solaris OS, 9
- 命名服务, 22
- 命名文件, 标准, 13
- 模板, 配置文件, 74
- 默认
 - 安全性配置文件, 29
 - 配置, FTP 和 Telnet, 17
- 模式, 32
- 目录
 - finish 脚本, 9
 - JumpStart 配置文件, 10
 - 结构, 4
 - 列表, 4
 - man, 5

OS, 9

- 启动, 7
- 驱动程序, 5
- 软件包, 10
- sysidcfg, 11
- 审计脚本, 4
- 文件, 8
- 修补程序, 10
- 运行, 57

N

- netstat 命令, 24
- NFS
 - 应用程序依赖于, 24
- NIS, 22

O

- o 选项, 撤消, 62
- o 选项, 审计, 84
- OEM Solaris OS 群集, SUNWCXall, 75
- oem.profile, 75
- OpenSSH
 - 编译, 40
 - 创建及配置, 40
 - 软件, 下载, 39
- OS
 - 目录, 9
- OS 群集, 指定和安装, 案例, 97
- OS 映像, 9

P

- pfiles 命令, 25
- pkgadd 命令, 36
- pkill 命令, 23
- pldd 命令, 20
- ps 命令, 23
- 配置
 - 安全性评估, 54

- JumpStart 服务器, 71
- JumpStart 服务器, 案例, 98
- JumpStart 模式, 72

- 检查准则, 54
- 监视和维护, 29
- 脚本, 10
- 配置环境, 31
- 评估, 案例, 108
- 审计, 80
- 审计报告, 88
- 信息, 驱动程序, 5
- 正在运行的和存储的之间的差异, 27
- 准则, 2
- 自定义, 案例, 94, 103
- 自动化, 2

配置文件

- 规划和准备, 15
- JumpStart, 10, 74
- JumpStart 配置文件, 10
- 检测, 91
- 目录, 10
- 确定是否正在使用, 21
- 修改, 74
- 主要, 5

- 屏蔽墙, 103
- 评估系统, 80
- 平台最小化, 21
- 破坏内容, 文件, 57, 58

Q

- q 选项, 撤消, 62
- q 选项, 审计, 84
- 启动目录, 7
- 嵌套或分层安全性配置文件, 26
- 强身份验证, 39
- 强制选项, 61
- 清单文件, 58
- 清单文件项
 - 处理多个, 65
- 情形, 保护系统安全, 93
- 驱动程序

- 命名, 13
- 目录, 5
- 配置信息, 5
- 驱动程序控制流程, 6
- 驱动程序目录, 5
- 驱动程序选项, 49
- 驱动程序, **JumpStart** 服务器, 73
- 权限
 - 对象, 默认, 38
- 权限管理, 16
- 权限, 保护, 38
- 确定保留为启用状态的 OS 服务, 53
- 确定动态加载的应用程序, 20

R

- rc 脚本, 审计运行, 80
- Recommended and Security Patch Clusters**
 - 存储, 10
 - 下载, 37
- reverse-jass-manifest.txt 文件, 58
- rm_install_client 命令, 77
- rm-client 脚本, 3, 77
- root
 - 的选项, 52
 - 目录, 36
- RPC**
 - 端口映射器, 22
 - 服务, 103
 - rpcinfo 命令, 22, 23
- rules 文件
 - JumpStart** 服务器, 73, 75
 - 检查, 案例, 101
- rusers 服务, 验证, 23
- rusers 命令, 23
- 日志
 - 操作, 57
 - 考虑, 15
- 日志文件
 - 安装, 28
 - 查看, 28

- 入侵检测, 16
- 软件安装, 脚本, 10
- 软件包
 - 目录, 10
 - 添加非 pkg 格式的软件包, 59
- 软件包名称, 案例, 105
- 软件包目录, 10
- 软件包, 添加非 pkg 格式的软件包, 59
- 软件组件, 3

S

- SCCS**, 11
- scp 命令, 37
- Secure Shell**
 - 安装, 案例, 105
 - 产品要求, 35
 - 创建及配置, 40
 - 软件, 下载, 39
 - 商业版本, 编译, 40
- secure.driver, 执行, 45
- SI_CONFIG_DIR, 在子目录中安装软件, 73
- SIGHUP 信号, 23
- SNMP**, 24
- Solaris Fingerprint Database Companion**, 41
- Solaris Fingerprint Database Sidekick**, 41
- Solaris OS**
 - 服务, 检查, 53
 - 命名标准, 9
 - 群集, SUNWCreq, 74
 - 软件包格式, 36
 - 修正, 37
 - 映像, 9
- Solaris Security Toolkit**
 - 软件, 下载, 36
 - 为 **JumpStart** 模式安装, 73
- Solaris 指纹数据库**, 41
- Solstice DiskSuite™**, 95
- SunSolve OnLine Web** 站点, 37
- SUNwjass 目录, 36
- SUNwjass-*n.n*.pkg, 36
- sysidcfg

- 目录, 11
- 文件, 73
- 文件范例, 11
- 文件, 版本限制, 72
- 文件, 为 JumpStart 模式做修改, 72
- syslog
 - 存储库, 29
 - 消息, 日志, 29
- 删除客户机, 从 JumpStart 服务器, 77
- 设计, Solaris Security Toolkit 软件, 1
- 审计
 - 案例, 108
 - 安全性评估, 89
 - 备份, 警告, 89
 - 标题, 85
 - 电子邮件选项, 83
 - 定制, 80
 - 过程, 91
 - 仅报告失败, 86
 - 控制输出, 82
 - 命令, 82
 - 排序输出, 88
 - 配置报告, 88
 - 日志项, 范例, 88
 - 输出选项, 84
 - 显示结果, 84
 - 消息, 85
 - 选项, 82
 - 抑制显示输出选项, 84
 - 周期性, 80
 - 主机名、脚本名称和时间戳信息, 88
 - 自动化, 79
 - 最小扫描, 80
- 审计, 限制, 1
- 审计, 已定义, 79
- 审计策略, 29
- 审计脚本
 - 定制, 81
 - 目录, 4
 - 所有权, 80
 - 相匹配的驱动程序, 58
- 审计系统, 79
- 审计选项, 46

- 生命周期, 维护安全性, 55
- 使系统安装标准化, 71
- 手动查看, 安全, 29
- 手动更改, 在撤消过程中保留, 62
- 守护程序, 禁用, 39
- 收集信息, 正在运行的进程, 19
- 输出
 - 禁用, 51
 - 排序审计, 88
 - 审计运行范例, 90
 - 最小化, 86
- 输出选项
 - 撤消, 62
 - 审计, 84
 - 文件, 51
- 数据完整性, 16
- 数字指纹, 40
- 私有管理网络, 103
- sun4u, 40
- 所有权的驱动程序和脚本, 80

T

- TCP 封装器, 105
- Telnet, 启用, 81
- truss 命令, 20, 29
- ttssession 进程, 25
- 特洛伊, 定义, 40
- 提取修补程序, 10
- 体系结构, Solaris Security Toolkit 软件, 4
- 添加 JumpStart 客户机, 案例, 99
- 添加客户机, 从 JumpStart 服务器, 76
- 停运时间, 16
- 通知, 在撤消过程中产生, 62
- 脱机, 保护系统安全, 16

U

- uncompress 命令, 36
- undo-log.txt 文件, 58

user.init 文件, 5
user.init.SAMPLE, 用途, 13
user.run.SAMPLE, 用途, 13

W

完整性

- 二进制文件, 检查, 40
- 可执行文件, 验证, 41
- 软件, 下载, 41
- 数据, 16
- 文件系统, 16

完整性管理解决方案, 11

维护安全, 29, 79

维护版本控制, 11

维护窗口, 16

稳定性, 37

文件

- 不一致, 62

- JumpStart** 客户机, 存储, 8

- 检查手动更改, 60

- 列出并检查更改, 59

- 命名标准, 13

- 目录, 8

- 配置文件, 74

- 破坏内容, 57, 58

- 确定使用情况, 25

- 修改, 13

文件范例, sysidcfg, 11

文件名, 36

文件系统

- 完整性, 16

文件系统对象

- 获得信息, 19

文件校验和, 59

问题, 28

X

系统

- 安全漏洞, 29

- 调用, 21

- 二进制, 验证, 41

- 配置, 监视和维护, 29

- 损坏, 57, 58

- 稳定性, 验证, 27

- 要求, 案例, 94

- 引导, 消息, 27

- 状态, 19

系统日志

- syslog.conf 文件, 自定义, 106

下载安全性软件, 35

显示帮助选项, 48

显示帮助选项, 审计, 82

限制编译器, 40

限制服务, 103

相关资源, xviii

详细级别, 84

消息, 审计, 85

校验和, 59

性能

- Solaris OS** 修补程序, 37

修补程序, 37

- 安装, 10

- 安装后重新加强系统, 32

- 创建子目录, 10

- 覆盖配置文件, 29

- 命名目录, 10

- 目录, 10

- README** 文件, 37

- 提取, 10

- 添加未安装的修补程序, 81

- 移动文件, 37

修改

- 代码, 12

- 配置文件, 74

修改, 跟踪, 57

修改, 验证, 53

需要的软件, 35

选项

- 备份, 撤消, 61

- 撤消命令, 61

- 电子邮件通知, 50

- 电子邮件, 撤消, 62

- 电子邮件, 审计, 83
- help, 48
- jass-execute 命令, 43
- 历史记录, 50
- 驱动程序, 49
- root, 52
- 审计, 46, 82
- 审计, 帮助, 82
- 输出文件, 51
- 抑制显示输出, 51
- 抑制显示输出选项, 撤消, 62
- 抑制显示输出, 审计, 84
- 最近执行, 50

Y

验证

- 安全性配置文件安装, 28
- 功能, 多次重新引导, 16
- 系统稳定性, 27
- 应用程序和服务功能, 28

验证, 在安装之前, 27

验证安全性配置文件, 55, 79

验证过程, 22

要求

- 安全性, 17
- 撤消加强运行, 58
- 服务, 16
- 服务, 确定, 18
- 收集, 21
- 应用程序, 16

异常行为, 22

移动修补程序文件, 37

依赖性

- 确定, 25
- 未识别的, 16

以太网接口, 案例, 95

抑制显示输出选项, 51

应用程序

- 标识, 16
- 清点, 18
- 确定动态加载的, 20
- 确定是否使用 RPC 端口映射器, 22

验证, 案例, 108

要求, 16

应用程序安全性, 16

应用程序启动, 消息, 27

应用修补程序, 29

用户交互式服务, 禁用, 39

用户交互式会话, 保护, 39

用途, Solaris Security Toolkit 软件, 1

预防措施, 15

源代码控制系统 (SCCS), 11

元服务, 22

运行目录, 57

Z

在独立模式下执行软件, 45

责任, 15

站点专用的驱动程序, 匹配审计脚本, 80

指定并安装 OS 群集, 案例, 97

质量保证 (QA) 测试, 54

执行 Solaris Security Toolkit, 42

注释标记 (#), 23

注释处理, 106

助手功能, 59

自定义配置, 案例, 94

自动审计, 79

最近执行选项, 50

最小化, Solaris 操作系统, 18

最小化输出, 86