

Automating UNIX and Linux Administration

UNIX 和 Linux 自动化管理

- ❖ 讲授如何自动执行日常的系统管理任务
- ❖ 研究自动化原理和流行的工具软件，阐述如何用 bash 和 Perl 开发自定义解决方案
- ❖ 探讨常见的程序，如 GNU cfengine、RPM、CVS、rsync、AutoRPM、Logwatch 和 OpenPKG

(美) Kirk Bauer 著
马孝荣 冯 锐 译



清华大学出版社

UNIX 和 Linux 自动化管理

(美) Kirk Bauer 著

马孝荣 冯锐 译

清华大学出版社

北 京

内 容 简 介

自动化系统管理技术可以帮助系统管理员从繁重枯燥的任务中得以解脱。本书利用目前流行的 bash、Perl、grep、sed 等工具软件详细地讲解了自动化系统管理领域最为实用的核心技术。主要内容涉及登录脚本、网络管理、自动化系统配置、数据共享、程序补丁、系统维护、系统监控、数据备份和系统安全等。

本书适合系统管理员以及准备学习系统管理知识的读者阅读。

EISBN: 1-59059-212-3

Automating UNIX and Linux Administration

Kirk Bauer

Original English language edition published by Apress L. P., 2560 Ninth Street, Suite 219, Berkeley, CA 94710 USA. Copyright ©2003 by Apress L.P. Simplified Chinese-Language edition copyright ©2004 by Tsinghua University Press. All rights reserved.

本书中文简体字版由 Apress 出版公司授权清华大学出版社出版。未经出版者书面许可,不得以任何方式复制或抄袭本书内容。

北京市版权局著作权合同登记号 图字: 01-2003-8757

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

UNIX 和 Linux 自动化管理/(美)鲍尔(Bauer,K.)著;马孝荣 冯锐译.—北京:清华大学出版社,2004

书名原文: Automating UNIX and Linux Administration

ISBN 7-302-08594-3

I. U… II. ①鲍… ②马… ③冯… III. ①UNIX 操作系统②Linux 操作系统 IV.TP 316.81

中国版本图书馆 CIP 数据核字(2004)第 041065 号

出 版 者: 清华大学出版社 地 址: 北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编: 100084

社 总 机: 010-62770175 客户服务: 010-62776969

组稿编辑: 曹 康

文稿编辑: 杜一民

封面设计: 康 博

版式设计: 康 博

印 刷 者: 北京市清华园胶印厂

装 订 者: 三河市李旗庄少明装订厂

发 行 者: 新华书店总店北京发行所

开 本: 185×260 印张: 26.75 字数: 684 千字

版 次: 2004 年 6 月第 1 版 2004 年 6 月第 1 次印刷

书 号: ISBN 7-302-08594-3/TP·6164

印 数: 1~4000

定 价: 54.00 元

前言

不可否认,因为您希望偷懒才会选择阅读这本书。偷懒的系统管理员是一些很出色的人——他们愿意执行一劳永逸的工作。早上一觉醒来,抓起桌边的笔记本电脑,查看电子邮件,然后一头冲向湖畔悠闲散步,这是我们所有人都梦想的生活。

使用本书的技术,您就能接近全自动化系统管理的理想世界。虽然经常会有不可预料的因素导致麻烦产生,我们至少能把一些重复的具有代表性的任务交给电脑去做(当然,发明电脑的初衷就是为了使我们的生活变得更加轻松)。至于如何让您的老板相信您,让您一周只需工作一天,这就要看您的了。

自动化的优越性

在大多数场合下,采用自动化的动机是为了节省时间。我们是忙人,我们的时间是有价值的。我们宁愿写一个脚本来添加用户,而不愿重复地手工添加用户。节省下来的时间可以处理不易自动完成的事情(或者是一些更有趣的事情)。当然,自动化系统管理的好处不止这些,还有一些好处不是这么明显。

在很多情况下,一些人因为没有足够多的相关知识而无法独立完成某些任务,而自动化方式则能为其代劳。在您手下工作的没有经验的系统管理员以及为公司提供技术支持的人员中都不乏其人。自动化让每一个人的生活变得更轻松。他们没有必要经常打扰您,而您也不用每天不得不回答同样的问题。

同等重要的是您采用自动化系统管理而自然形成的文档。比如,为了增加一个新账号,除了在文件服务器里创建该账号的 home 目录,设置自动挂载表外,还得把该账号添加到 passwd 文件、shadow 文件以及 group 文件。在绝大多数管理任务中,您通常会很详细地编制文档,遵循循序渐进的操作步骤,但有时您也会一时疏漏某项特定的任务。

这就是自动化管理最有意义之处。如果您编写了一个脚本来完成添加用户所要求的所有操作,就相当于已经卓有成效地写好了一个操作指南,阐述如何创建一个新账号。如果在脚本里添加了一些有用的注释,也就同时为操作编写了文档。即使您 3 个月以来没有添加任何新账号,这个脚本仍然唾手可得的。即使假设这个脚本已经过时了,在运行当中会出现错误,您依然知道脚本中的哪些操作可能存在问题,哪些操作可能运行正常。只要对这个脚本稍加修改就能让它继续工作。

使用自动化技术

即使限制在单系统环境下,自动化技术依然是有益的。不管是多么简单的系统,也不管是

多么微小的系统，繁琐的日常操作仍然需要去做，日志仍然需要被监视。当然，在跨系统部署时，自动化管理才真正体现出它独有的优越性来。

管理成千上万台机器是一件很惬意的事情，然而如果处理不当的话，这也会是一场噩梦。为了避免噩梦的发生，减轻以后的工作负荷，请确保事情从开始就是正确的。当然，如果管理的只是现有的几台机器，为了让您的生活变得更轻松，同样有很多可做的事情。因此就算您不能从头开始，您依然会发现本书是大有帮助的。比如，您可以使用一些方法让您现有网络中的机器自动实现标准化。

如果您管理着多台机器，可能您会想，如果所有的机器有着同样的硬件、操作系统和软件该有多好。有时候这是可能的(例如，如果您正在使用 Beowulf 簇的话)；不过通常这是不可能的(尤其是在有着 1,000 多个程序员的软件公司里)。本书同时适用于一致的和混合的 Unix 环境。虽然我所举的例子可直接应用到 Linux 和 Unix 环境，但是您可以把这些思想运用到任何系统环境中。

谁将从中受益

这本书是写给有经验的系统管理员的，这并不是说您必须是一个专家才行，而是说在您有了一定的经验之后，才会发现本书的真正价值所在。例如，如果不查找参考手册的情况下，您记不起来如何挂载一个文件系统，那么本书对您来讲也许难度大了点。

除了假设您能够执行一些基本的系统管理任务外，还假设您能熟练使用 Perl 和/或 bash 脚本。另外，希望您能理解基本的正则表达式语法。

如果您是学生，正在运行 Linux 服务器，那么本书是适合您的。如果您在管理几台 Unix 系统，那么本书也会对您有所帮助。如果您管理着一些 Unix 工作站，一组 Unix 服务器，一个 Web 群，或者一个 Beowulf 簇，相信本书将成为您的良师益友。

本书涵盖的技术

本书阐述了 3 种解决自动化问题的途径，在实际操作中您可以选用其中一种：

开放源代码软件：开放源代码解决方案可以解决本书所阐述的大多数问题，但是我不会涉及开放源代码方案所能解决的每一个问题。我将分类阐述最流行且最为成熟的软件。它们包括 GNU cfengine、Logwatch、Swatch、AutoRPM 以及 NetSaint(该程序当前作为 Nagios 正在开发中)。本书不是其中任何一个软件的完全教学手册，但是它确实提供了足够多的信息来帮助您考虑和着手使用这些程序。就在本书中提供：

定制脚本和软件：本书的另一个焦点是定制解决方案。既然开放源代码解决方案已经存在，为什么我还要提倡定制解决方案呢？如果可能的话，我建议您使用现有的解决方案。然而，在很多场合下，现有的解决方案过于复杂，功能却不够强大，或者对您特定的需求来说限制太多。而且，有如此多的自动化管理领域并没有现成的解决方案。这通常是因为每个案例的情况都会有明显的差异，或者您所需求的解决方案过于简单，难以为此开发专门一种产品。

商业软件：本书没有介绍商业软件——并不是因为软件本身有问题，而是因为在很多场合下应用商业软件成本太高(尤其是考虑与软件相关的咨询成本时)。而且，我也无法获得足够的与商业解决方案相关的信息来进行全面的论述。当然，在一些场合下，商业软件也许是最好的解决方案，我建议您考虑它是否适合您的需要。

章节摘要

本书首先作章节介绍，您应该熟记各章的主要内容。然后本书的每一个章节讨论自动化管理的一个核心领域。各章的论述全面深刻，同时讨论现有和定制的方案，还提供了非常多的实例。

第1章：自动化管理基础。本章介绍了系统自动化管理背后的方法论。

第2章：使用 SSH 安全地进行自动化系统管理。本章介绍了使用 Secure Shell(SSH)的基本方法，讨论了 SSH 安全所涉及到的问题，描述了如何设置无口令的 SSH，还探讨了各种各样相关主题。

第3章：创建登录脚本和 Shell 脚本。本章讨论了 bash 程序的一些较高级的特征以及如何使用它们去定制提示符，创建命令别名，增强 tab 键的自动补齐功能，进而提高您的 shell 编程经验。本章还提供了一些允许跨系统运行命令的脚本。

第4章：安装之前：网络准备和管理。本章讨论在网络中添加新系统需要完成的任务，还提供了定制的一个模块化的 shell 脚本来自动完成这些任务。

第5章：自动安装和定制安装。本章讨论了自动执行操作系统安装时可用的一些选项。还讨论了定制操作系统时的可用选项，并提供了一些脚本，允许您快速轻松地将一个新系统(甚至是未经定制的系统)添加到自动化系统中。

第6章：自动化系统配置。本章介绍了网络中所有系统的自动化配置，不考虑正在运行的操作系统类型或者它们所执行的任务类型。本章提供了定制解决方案以及关于 GNU cfengine 应用的综合论述。

第7章：在系统间共享数据。本章讨论了在不同的系统间共享数据的多种可用方法以及网络文件系统，如网络文件系统(NFS)、网络信息服务(NIS/NIS+)、GNU cfengine、协作版本系统(CVS)以及超文本传输协议(HTTP)和文件传输协议(FTP)。

第8章：程序包和补丁。本章讨论了与系统更新相关的多个主题。内容涵盖了 Solaris 和定制补丁，Red Hat 包管理器(RPM)和 Debian 包格式，以及一个定制程序包解决方案。此外，本章还阐述了使用 AutoRPM 来自动化程序包安装，介绍了 OpenPKG 系统，该系统提供的程序包能安装到很多不同的操作系统中。

第9章：系统维护和更改。本章讨论了存在于绝大多数现代操作系统中的各种各样的维护任务。这些任务包括时间同步、账号管理、日志文件重置以及日常系统清理。

第10章：系统监控。本章介绍了监控系统的技术。还讲解了使用 Logwatch 和 swatch 来进行日志监控，使用 NetSaint 来进行网络监控。

第11章：提高系统的安全性。本章讨论了一些用于增强系统安全同时降低维护工作量的工具。本章介绍了 Tripwire 的用法和配置，如何创建自动化的防火墙配置，以及如何使用 GNU

cfengine 来增强系统和网络的安全性。

第 12 章：数据的备份和恢复。本章介绍了一些简单廉价的解决方案，来帮助您自动备份系统中的数据。

第 13 章：用户界面。本章介绍了如何使用 bash、Perl 和 Mason 来为自动化系统创建控制台和基于 Web 的用户界面。

附录 A：基本工具介绍。介绍了贯穿本书的基本工具，为您理解和应用本文中的例子提供了一个良好的起点。本附录涵盖了如下工具：bash、Perl、grep、sed 和 awk。

附录 B：定制和自动化 Red Hat Linux 安装。介绍了如何自动化安装 Red Hat Linux。本附录还介绍了如何根据 Red Hat Linux 来创建定制的 Linux 发布程序。

附录 C：构建 Red Hat 程序包管理器(RPM)的程序包。介绍了如何建立自己的 RPM 程序包。

附加资源

自动化系统管理涵盖了一个很广泛的范围。本书推荐了一些附加的阅读资料。此外，还有几本书非常有帮助，下面介绍给大家：

- **UNIX System Administration Handbook (第 3 版)**, 作者: Evi Nemeth, Garth Snyder, Scott Seebass, Trent R. Hein (Prentice Hall, 2000 年)。本书几乎涵盖了所有关于 Unix 系统管理的资料。
- **Principles of Network and System Administration**, 作者: Mark Burgess (John Wiley & Sons, 2000 年)。本书讨论了优秀的网络和系统管理中的一些理论和原则。
- **Unix Storage Management**, 作者: Ray A. Kampa 和 Lydia V. Bell (Apress, 2002 年)。本书覆盖了关于 Unix 系统数据存储的方方面面。
- **Learning the bash Shell**, 作者: Cameron Newham 和 Bill Rosenblatt (O'Reilly and Associates, 1998 年)。本书涵盖了 bash 命令 shell 和 bash shell 脚本。
- **Learning Perl**, 作者: Randal L. Schwartz 和 Tom Phoenix (O'Reilly and Associates, 2001 年)。本书是学习 Perl 的好资料。Perl 是一种功能非常强大的脚本语言，本书深入探讨了它的用法。
- **Programming Perl**, 作者: Larry Wall 等 (O'Reilly and Associates, 2000 年)。这是一本更为高级的 Perl 指南，提供了广泛的参考资料。
- **Perl for System Administration**, 作者: David N. Blank-Edelman (O'Reilly and Associates, 2000 年)。上面已提到，在本书中我们将广泛地使用 Perl。顾名思义，就知道这本书是关于自动化系统管理的。推荐本书的意图也就很明确了。
- **Practical UNIX & Internet Security**, 作者: Simson Garfinkel, Alan Schwartz 和 Gene Spafford (O'Reilly and Associates, 2003 年)。本书涵盖了应该时刻铭记在心的安全问题，尤其是在进行自动化系统管理的时候。

本书约定

本书使用了多种不同的字体和格式，本节对这些约定和它们的用法作一个简介：

注释：

这个标记表明给定的论题中有一些重要和有趣的观点需要注意。

提示：

提示会为您提供该主题的一些额外信息，您会发现它们很有用。

小心：

小心用来警告您所执行的操作或者提供给您的脚本可能会引起的潜在的问题。

警告：

警告提供的重要信息是您在继续操作之前必须要提防的，比如较为严重的潜在问题。

本书包含了大量的程序清单，有时候它们以代码块的形式显示出来。

```
#!/bin/bash
echo "hello, how are you?"
```

有时候它们可能会被分成几部分。例如会首先定义一个函数：

```
say_hello() {
    echo "Hello, how are you?"
}
```

然后提供调用这个函数的代码。

```
case $0 in
    hello)
        say_hello
        ;;
    esac
```

为了运行这些代码，可以从 Apress 网站 (<http://www.apress.com>) 的下载页面去获取完整的代码。还可以把这些代码段拼凑成一个文件来创建自己的脚本，在文件的头部加上适当的解释器声明(例如#!/bin/bash)。

最后，一些代码清单会被标上行号，一些文本行会被加粗。这些行在代码后将被更详细地讨论。下面是一个示例：

```
1 #!/bin/bash
2
3 echo "Hello, how are you?"
```

第 1 行：使用 `bash` 解释器运行代码。

第 3 行：使用 `echo` 命令通过 `Stdout` 输出字符串。

您会在书中经常看到一些命令运行的示例，如下所示：

```
% ./hello
```

```
echo "Hello, how are you?"
```

```
#!/bin/bash
```

```
echo "Hello, how are you?"
```

目 录

第 1 章 自动化管理基础	1
1.1 是否需要自动化管理	1
1.1.1 采用多种系统的大型公司	2
1.1.2 成长中的中型公司	2
1.1.3 Internet 服务提供商	2
1.1.4 应用服务提供商	2
1.1.5 Web 服务器群	2
1.1.6 Beowulf 簇	3
1.1.7 网络设备	3
1.2 阅读本书的收获	3
1.3 系统管理员的任务	4
1.4 提供用户界面	4
1.5 选用正确的方法	5
1.5.1 均衡系统	6
1.5.2 推和拉	6
1.6 处理用户和管理员的关系	7
第 2 章 使用 SSH 安全地进行自动化系统管理	10
2.1 SSH 基础	10
2.2 使用 SSH 增强安全性	11
2.3 使用 RSA 认证	12
2.3.1 生成密钥对	12
2.3.2 指定授权的密钥	13
2.4 使用 ssh-agent	14
2.4.1 ssh-agent 的基本用法	14
2.4.2 ssh-agent 的高级用法	15
2.4.3 密钥转发	16
2.5 限制 RSA 认证	17
2.5.1 处理未获信任的主机	18
2.5.2 允许执行权限受限的命令	18
2.5.3 端口转发	18
2.6 对公共账户使用 SSH	19

2.6.1 准备建立公共账户	19
2.6.2 监控公共账户	23
第 3 章 创建登录脚本和 shell 脚本	25
3.1 定制命令提示符	25
3.1.1 让提示符提供更多信息	25
3.1.2 使用有色的提示符	26
3.2 使用 Tab 补齐功能	29
3.2.1 配置 Tab 补齐功能	30
3.2.2 可编程的 Tab 补齐功能	30
3.2.3 使用定制的补齐函数	30
3.3 创建命令别名	31
3.3.1 使用别名提高效率	32
3.3.2 提高别名的一致性	32
3.4 创建影响多个系统的命令	35
3.4.1 简单的 Multi-SSH	35
3.4.2 基于组的 Multi-SSH	36
3.5 分发登录脚本	38
3.5.1 跟踪不同版本	39
3.5.2 分发登录脚本	39
第 4 章 安装之前：网络准备和管理	42
4.1 确定使用静态 IP 还是动态 IP	42
4.2 创建和使用模块化 shell 脚本	43
4.3 使用锁定功能来防止冲突	43
4.4 基本的预安装脚本	44
4.5 预安装组件脚本	47
4.5.1 请求 MAC 地址	48
4.5.2 分配 IP	49
4.5.3 对系统分类	50
4.5.4 收集联系人信息	52
4.5.5 添加到/etc/hosts 文件中	52
4.5.6 配置 DHCP 服务器	53
4.5.7 更新 DNS 配置	54
4.6 如何处理错误	56
4.7 移除机器	57
4.7.1 移除 MAC 地址	58
4.7.2 回收 IP	59
4.7.3 撤消系统	59

4.7.4	删除联系人条目	60
4.7.5	修改主机文件	60
4.7.6	配置 DHCP	61
4.7.7	更新 DNS 配置	62
4.8	生成报告	63
4.8.1	有多少台机器	63
4.8.2	可用的 IP 地址有多少	64
4.8.3	正在使用哪些 MAC 地址	64
第 5 章	自动安装和定制安装	65
5.1	自动化安装	65
5.2	定制操作系统	66
5.2.1	是否应该定制操作系统	66
5.2.2	维护定制发行软件	67
5.3	准备进行自动化管理	67
5.3.1	远程配置新系统	68
5.3.2	自我配置的新系统	73
第 6 章	自动化系统配置	76
6.1	配置内容	76
6.1.1	定制文件	76
6.1.2	动态文件	77
6.1.3	默认文件	77
6.1.4	静态文件	77
6.1.5	文件系统组件	77
6.2	遵循的配置原则	78
6.3	创建定制配置方法	78
6.3.1	定义文件和目录结构	79
6.3.2	定义配置逻辑	82
6.3.3	提供配置脚本	82
6.3.4	分析示例配置文件的创建	88
6.4	使用 GNU cfengine 配置系统	90
6.4.1	cfengine 概述	90
6.4.2	基本安装	94
6.4.3	调试 cfengine	99
6.4.4	在 cfagent.conf 中创建代码段	100
6.4.5	使用 cfrun	105
第 7 章	在系统间共享数据	107
7.1	数据类型	107

7.1.1	共享应用程序	108
7.1.2	用户数据	108
7.1.3	应用程序数据	109
7.1.4	系统信息	109
7.1.5	账号信息	110
7.2	使用网络文件系统	110
7.2.1	使用 NFS 共享数据	110
7.2.2	使用 Samba 共享文件	112
7.2.3	使用 Andrew 文件系统(AFS)	113
7.2.4	其他可用的网络文件系统	113
7.3	自动挂载网络文件系统	114
7.3.1	Solaris 的自动挂载程序	115
7.3.2	Linux 上的自动挂载程序(autofs)	116
7.3.3	am-utils 的自动挂载程序	116
7.4	使用 NIS/NIS+共享系统数据	118
7.4.1	创建 NIS 映射和 NIS+表	119
7.4.2	/etc/nsswitch.conf 文件	120
7.4.3	netgroup 文件	121
7.4.4	NIS 细节	121
7.4.5	使用 NIS 的安全性问题	122
7.4.6	NIS+细述	122
7.5	使用轻量级目录访问协议(LDAP)	123
7.6	Kerberos 的安全性	123
7.7	使用 cfengine 共享数据	124
7.7.1	分发文件	124
7.7.2	管理 NFS 的挂载	125
7.8	用 rsync 同步数据	129
7.8.1	rsync 的潜在用途	130
7.8.2	决定使用哪一种 rsync 传输协议	131
7.8.3	rsync 基本应用	131
7.8.4	rsync 应用实例	133
7.9	用 CVS 管理和分发数据	135
7.9.1	CVS 基础	135
7.9.2	系统管理脚本的分发	141
7.9.3	Web 内容分发	144
7.10	用 HTTP/FTP 传输数据	145
7.10.1	实例: Web 内容同步	146
7.10.2	简单实例: 传输系统配置文件	146

7.10.3 进阶实例：传输系统配置文件	147
7.11 RPM	154
第 8 章 程序包和补丁	155
8.1 补丁和程序包	155
8.2 如何选择	156
8.3 如何开始	157
8.4 利用补丁更新系统	158
8.4.1 理解和应用 Solaris 补丁	158
8.4.2 定制补丁	164
8.5 理解和安装程序包	176
8.5.1 Red Hat 程序包管理器(RPM)	177
8.5.2 Debian 程序包	187
8.5.3 Slackware 程序包	193
8.6 利用 AutoRPM 实现程序包的自动安装	197
8.6.1 基本的 AutoRPM 设置	198
8.6.2 在交互模式中工作	199
8.6.3 理解 FTP 池文件	200
8.6.4 配置文件示例	200
8.6.5 AutoRPM 触发器	207
8.7 利用 OpenPKG 执行跨平台打包操作	208
8.7.1 安装 OpenPKG	209
8.7.2 利用 OpenPKG 安装程序包	210
8.7.3 修改程序包	211
第 9 章 系统维护和更改	213
9.1 同步时间	214
9.1.1 设置时区	214
9.1.2 同步时钟	214
9.1.3 更新硬件时钟	215
9.2 管理账户	215
9.2.1 设计账户管理脚本	216
9.2.2 安排账户管理配置文件	217
9.2.3 账户管理辅助函数	218
9.2.4 账户管理主脚本	220
9.2.5 账户管理数据组件	222
9.2.6 修改组件	227
9.2.7 清除已删除用户	233
9.3 维护日志文件	235

9.3.1	Red Hat 的 logrotate	235
9.3.2	利用 spinlogs 轮循日志	237
9.3.3	利用 cfengine 进行日志轮循	238
9.4	删除文件	238
9.4.1	定制驱动器清理	239
9.4.2	Red Hat 的 tmpwatch	240
9.4.3	利用 cfengine 删除文件	241
第 10 章	系统监控	243
10.1	常规系统监控	243
10.1.1	创建常规报告工具	244
10.1.2	监控系统负载	246
10.1.3	监视可用的磁盘空间	247
10.1.4	监控系统服务	248
10.1.5	监视程序包的变化	250
10.1.6	驱动器故障	251
10.2	监控系统日志	253
10.2.1	利用 Logwatch 进行日志监控	253
10.2.2	利用 swatch 进行实时日志监控	259
10.3	监控网络服务	260
10.3.1	自定义监控和自动修复	261
10.3.2	NetSaint(又名 Nagios)	263
10.3.3	Mon	272
第 11 章	提高系统的安全性	276
11.1	利用 GNU cfengine 提高系统安全性	276
11.1.1	执行基本的文件检查	277
11.1.2	disable 段	277
11.1.3	files 段	278
11.1.4	控制网络服务	280
11.2	配置系统级的防火墙	282
11.2.1	防火墙配置文件	283
11.2.2	防火墙配置脚本	283
11.2.3	系统组	285
11.3	利用 Tripwire 监视文件	286
11.3.1	安装 Tripwire	287
11.3.2	生成 Tripwire 策略文件	287
11.3.3	使用 Tripwire	291

第 12 章 数据的备份与恢复	294
12.1 确定备份策略	294
12.2 使用 cfengine 备份数据	296
12.2.1 使用 cfengine 执行本地备份	296
12.2.2 在备份时排除文件	297
12.3 使用 rsync 备份数据	297
12.3.1 使用 rsync 进行基本的备份	298
12.3.2 增量备份	298
12.3.3 排除文件	299
12.4 使用 rdiff-backup 备份数据	300
12.4.1 使用 rdiff-backup 进行增量备份	300
12.4.2 删除旧的备份数据	301
12.4.3 恢复数据	303
12.4.4 在使用 rdiff-backup 备份时排除文件	303
12.5 使用 tar 进行磁带备份	304
12.5.1 使用 tar 进行基本的备份	305
12.5.2 使用 tar 创建增量备份	305
12.5.3 磁带的备份	306
12.5.4 使用 tar 进行备份时排除文件	307
12.5.5 从 tar 包中恢复数据	307
12.6 Amanda 备份工具	308
第 13 章 用户界面	309
13.1 使用 Webmin 实现基于 Web 的管理	309
13.2 创建底层工作脚本	310
13.3 以 root 用户身份执行命令	313
13.4 设计并使用通用的数据文件	316
13.4.1 保存设置的通用文件格式	316
13.4.2 定义界面的规范	318
13.5 基于文本的界面	320
13.5.1 菜单界面	321
13.5.2 配置基于文本的界面	322
13.6 创建基于 Web 的界面	326
13.6.1 组织 Web 界面	327
13.6.2 Mason 简介	327
13.6.3 HTTP 认证的安全性	332
13.6.4 记录 Web 服务器的操作	334
13.6.5 创建并使用定制的 Perl 库	335
13.6.6 基于 Web 的系统总体状态信息	337

13.6.7	系统配置的 Web 界面	340
13.6.8	在 Web 上执行 shell 命令	342
附录 A	基本工具介绍	351
附录 B	RedHat Linux 的定制和自动安装	367
附录 C	构建 Red Hat 程序包管理器(RPM)的程序包	379