UNIX系统的安全与防卫

方毅铭　延伟　编译

北京航空航天大学出版社

UNIX系统的安全与防卫

UNIX 系统的安全与防卫

方毅铭
延　伟　　编译

北京航空航天大学出版社

(京) 新登字 166 号

内 容 简 介

本书从实践出发，全面、系统而具体地介绍了几种 UNIX 版本有关安全管理方面的机制，并从实用的角度指导读者如何保护数据与文件、如何保护系统不受各种人为的破坏、如何检查系统是否受到攻击并清除系统的破坏者。具体内容包括：

- 操作系统的安全等级
- 普通用户及系统管理员的安全管理及操作
- 通讯、网络的安全性问题
- 系统防卫
- UNIX 系统安全问题的未来发展
- 用于安全管理的实用程序

本书适用于在 UNIX 系统上工作的各种用户，同时对致力于研究 UNIX 操作系统安全性问题的学者以及正在学习 UNIX 操作系统的人来说，本书是一本难得的好书。

UNIX 系统的安全与防卫
UNIX XITONG DE ANQUAN YU FANGWEI

方毅铭 延 伟 编译
责任编辑 樊 毅
北京航空航天大学出版社出版
新华书店总店科技发行所发行 各地新华书店经销
朝阳科普印刷厂印装
787×1092 1/16 印张: 12.75 字数: 326 千字
1992 年 2 月第一版 1992 年 2 月第一次印刷 印数: 10000 册

ISBN 7-81012-291-6/TP·057 定价: 7.00 元

前　言

随着 UNIX 多用户操作系统广泛地进入计算机市场，并应用于各个领域，UNIX 系统的安全问题已成为一个非常严肃的问题，加之近几年来不断发生 UNIX 系统遭受攻击的触目惊心的事件，人们更感到此问题的严重性和迫切性。

本书主要围绕着当今比较流行的几个 UNIX 操作系统版本（BSD 版本、System V 版本、以及新发行的 SVR4.0 版本）对有关安全性问题作了非常全面的介绍和讨论。阅读此书不仅有助于对 UNIX 操作系统的安全机制作全面的了解，而且还有助于 UNIX 用户在实际工作中保护数据、保护系统、防御攻击。

本书主要参考了“UNIX SYSTEM SECURITY”一书。该书作者 Rik Farrow 多年在 UNIX 系统上从事系统管理员和编程工作，并致力于研究 UNIX 系统的安全性问题，该书是他多年来实践的总结，也是目前唯一一本专门介绍 UNIX 系统安全问题的技术性书籍。除此之外，本书还参考了 SUN OS 系统管理员手册和 SVR4.0 系统管理员手册的有关内容。

本书共分六章及附录部分，其中第一、四、五、六章由方毅铭编译，第二、三章及附录部分由延伟编译，全书的审校工作由方毅铭、延伟共同完成。由于水平有限，加之时间仓促，书中难免会有谬误之处，欢迎广大读者给予指正。

编译者

一九九一年十二月

目　录

第一章　UNIX 系统安全性引论 …… (1)
1.1 引论 …… (1)
1.2 UNIX 系统及其声誉 …… (1)
1.3 计算机安全性 …… (3)
1.3.1 必要的安全性 …… (4)
1.3.2 计算机安全性与道德行为 …… (4)
1.3.3 计算机安全性的一个模型 …… (5)
1.3.4 可靠计算基 …… (6)
1.4 橘皮书 …… (7)
1.4.1 橘皮书的定义 …… (9)
1.4.2 层次与级别 …… (10)
1.5 小结 …… (11)

第二章　普通用户的安全管理 …… (13)
2.1 口令 …… (13)
2.1.1 选择一个好口令 …… (14)
2.1.2 改变口令 …… (15)
2.1.3 保护终端不被利用 …… (16)
2.2 存取控制 …… (17)
2.2.1 文件的权限 …… (18)
2.2.2 改变权限 …… (19)
2.2.3 新文件的所属权 …… (21)
2.2.4 缺省的文件权限 …… (22)
2.2.5 目录权限 …… (23)
2.3 启动文件 …… (27)
2.3.1 良好的路径 …… (28)
2.3.2 启动文件的保护 …… (30)
2.4 更正权限 …… (32)
2.5 文件加密 …… (35)
2.6 小结 …… (38)

第三章　系统管理员的安全管理 …… (40)

3.1 系统管理员与安全管理员 …… (40)
3.1.1 系统管理员的安全管理职责 …… (40)
3.1.2 共享账户 …… (45)
3.1.3 非活动账户 …… (45)
3.1.4 账户口令 …… (46)
3.1.5 检查口令文件 …… (47)
3.1.6 加密的口令 …… (50)
3.2 系统所属的文件和目录 …… (57)
3.2.1 系统目录的写权限 …… (60)
3.2.2 系统文件的写权限 …… (64)
3.2.3 系统文件的读权限 …… (67)
3.2.4 设备文件 …… (68)
3.2.5 建立系统文件数据库 …… (70)
3.2.6 对照数据库文件检查系统 …… (73)
3.3 调整用户特权与调整组特权 …… (78)
3.3.1 授予用户特权 …… (78)
3.3.2 问题 …… (79)
3.3.3 解决方法 …… (81)
3.4 小结 …… (83)

第四章 通信与网络的安全性问题 …… (84)
4.1 延伸了的访问 …… (84)
4.2 modem 的不安全性 …… (84)
4.2.1 Modem 端口的配置 …… (85)
4.2.2 拨号口令 …… (86)
4.2.3 特殊的 modem 硬件 …… (87)
4.3 UUCP 安全性 …… (89)
4.3.1 版本 2 UUCP 的安全性问题 …… (90)
4.3.2 Honey DanBer 安全性问题 …… (92)
4.4 网络安全性 …… (99)
4.4.1 网络的构成 …… (100)
4.4.2 可靠主机 …… (101)
4.4.3 其他网络服务 …… (106)
4.4.4 关闭网络监视程序 …… (110)
4.5 受限环境 …… (111)
4.5.1 特殊的注册程序 …… (112)
4.5.2 受限 Shell …… (112)
4.5.3 改变根目录环境 …… (114)

第五章　系统防卫 …… (120)
5.1 问题的提出 …… (120)
5.2 观察系统的工具 …… (121)
5.2.1 who 命令 …… (121)
5.2.2 进程状态 …… (122)
5.2.3 whodo 及 w 命令 …… (124)
5.2.4 进程记账 …… (125)
5.2.5 日志文件 …… (128)
5.3 处理入侵事件 …… (129)
5.3.1 正在进行的入侵 …… (130)
5.3.2 过去的入侵事件 …… (132)
5.3.3 找出隐藏文件 …… (136)
5.3.4 检查可执行程序 …… (137)
5.4 法律起诉 …… (137)
5.4.1 告诫信息 …… (137)
5.4.2 上机条例 …… (138)
5.4.3 收集证据 …… (138)
5.4.4 其他 …… (139)
5.5 病毒与 UNIX 系统 …… (140)
5.5.1 PC 病毒 …… (141)
5.5.2 UNIX 病毒机制 …… (141)
5.5.3 避免 UNIX 系统感染病毒 …… (142)
5.5.4 检查 UNIX 的病毒 …… (143)
5.5.5 从病毒的攻击中恢复 …… (145)
5.6 小结 …… (146)

第六章　UNIX 安全性的未来 …… (148)
6.1 未来从现在开始 …… (148)
6.2 隐蔽口令文件 …… (149)
6.3 Sun OS 4.0 中的审计 …… (151)
6.3.1 审计机制 …… (152)
6.3.2 Sun OS 的审计配置文件 …… (153)
6.3.3 检查审计记录 …… (156)
6.4 B 级系统的强制性控制 …… (159)
6.4.1 System V / MLS …… (159)
6.4.2 其他 B 级安全性的特性 …… (160)
6.5 回顾橘皮书 …… (162)
6.6 小结 …… (162)

附录 A　lock shell 命令程序与 C 语言程序 …… (164)
附录 B　一个简单的加密程序 …… (168)
附录 C　不可靠口令字典 …… (170)
附录 D　口令时效的警告程序 …… (173)
附录 E　具有调整用户及组标识方式的程序清单 …… (175)
附录 F　如何编写一个具有调整用户标识方式的程序 …… (179)
附录 G　介绍 Internet Worm …… (188)

第一章 UNIX 系统安全性引论

1.1 引 论

UNIX 作为一个多用户操作系统目前已超出了最初的教育领域而广泛地应用于其他各个领域，如商业、政府和军事部门。随之而来的问题就是如何提高 UNIX 系统的安全性。这一问题已引起了计算机界及其应用领域中的广大工作者的密切关注。1988 年 11 月，Morris Worm 成功侵入 Internet 网这一惊人事件更使人们认识到了提高 UNIX 系统安全性的迫切性，从而把这一课题推向前沿。

1.2 UNIX 系统及其声誉

UNIX 系统在安全性方面曾一度声名狼藉，这一方面是由于象 Morris Worm 这样的病毒成功地侵入了 UNIX 系统；另一方面是由于 UNIX 系统最初大量地在教育领域使用，出于教育领域特有的背景，安全性问题一直未给予足够的重视。使 UNIX 安全性成为某些人笑料的原因不仅仅是由于系统设计上的许多固有的缺陷，而主要是由于 UNIX 系统的使用方法。

追朔 UNIX 系统的历史，就可寻找出它在安全性方面获得坏名声的根源。UNIX 最初是作为一个单用户的编程平台诞生于 Bell 实验室。作为 UNIX 系统的先辈，Multics 优先考虑了安全性，至今 Multics 仍是几个具有最高安全级的操作系统之一，UNIX 系统虽在有些概念上基于 Multics，但却大大简化了设计。UNIX 几乎是在一夜之间从一个单用户系统转变为一个多用户系统。

随着越来越多的文章介绍和专题讲授这个简单、清晰而实用的系统，大专院校开始收集并获得了这个系统的源程序。在 UNIX 系统诞辰后的 6 年中，出版了二本书（UNIX 操作系统源码第 6 版和 UNIX 操作系统评价，作者 J.Lions，Wales 大学，1977 年），书中提供了加了注释的 UNIX 系统核心源码，而这正是安全性机制的心脏部分。这两本书在 UNIX 编程人员中成了很畅销的读物。（这里的源码是指用来构造 UNIX 系统的 C 语言的文件与列表。有了源码，就象有了银行珍宝室的蓝图。程序员就可以了解 UNIX 核心中的缺陷和弱点，并利用这些缺陷和弱点。）这两本书如今已不太有用了，因为自 1977 年这两本书发行以来，UNIX 系统的核心已做了非常大的改动。

相比之下，IBM 大型机源码珍藏于一个很小的图书馆中。每本文档的封面上都有醒目的警告信息，以警告那些非法阅读者。这种警告在文档中还不断出现。所以，要想获得有关 IBM 操作系统的内部资料是很困难的，而且也不鼓励任何人搜寻这方面的资料。如今，UNIX 的源码已受到法律的保护，然而，UNIX 的内部情况还是可以通过阅读一些书籍和资料来获取到的。

“视 UNIX 为不安全系统”在某些方面来看是不公平的。有些大机器在出售时，其缺

省的安全性比一台运行低版本的 Xenix 微机系统的安全性还差。某家最大的小型机生产厂商在培训场地工程师时，在教他们用口令获取系统特权的同时还教他们不用口令而获取系统特权的方法。还有，最常用的 PC 计算机可以说是没有一点安全性可言。但不知为什么，只有 UNIX 被标为不安全的操作系统。

UNIX 的编程人员所构成的团体是一个开放式的团体。在这个团体中，经常召开会议以交流经验共享信息。这种开放式的气氛渗透到许多计算机环境中，在那里，任何用户可涉及任何文件或资源。虽然程序员喜欢在一个无安全性的系统上工作，但若把无安全性保障的系统推向商业领域，或把它们联入网络，则这样的系统是无法正常工作的。而许多情况恰恰是这样的，即把程序员使用的广泛开放的系统作为新的 UNIX 系统发行出去。对于一个 UNIX 系统安装的新手来说，他们只会按照安装手册上的指令来安装系统，而这些安装指令并不会增加系统的安全性。而对于编制系统软件的程序员来说，他们并不关心系统的安全性，因为他们的编程环境不需要什么安全性要求。

现在，人们越来越意识到系统安全的必要性。大多数 UNIX 系统在发行时有较严格的配置，并在系统安装时加入了一些维护系统基本安全性的指令。但有关如何维护 UNIX 系统的安全这方面的书籍和资料目前还比较缺乏。

本书的目的就是帮助读者了解有关 UNIX 系统安全性的特性和有关安全性问题。90%以上的 UNIX 安全性问题是由于用户和系统管理员的疏忽及错误而造成的。本书将使读者学会如何避免这些疏忽和错误。本书还帮助读者了解系统“破坏者”是如何突破系统的安全性从而攻击系统的。系统管理员掌握了这方面的情况后，即使不能阻止“破坏者”的攻击行为，至少可以检测到这种行为。无论对 UNIX 系统的新手，还是对富有经验的老手，本书都会有你想了解的内容。

本书的结构

第一章介绍背景以帮助读者了解计算机安全性的有关知识。这些背景知识将在后面章节中引用到，所以这一章实际上提供了一些重要的概念性的框架。

第二章涉及用户方面的问题，即每个使用 UNIX 系统的人必须知道和遵循的原则。对于一个安全的 UNIX 系统的用户，而不是系统的管理员和编程人员，则第一章和第二章的内容就够用了。对于某些高级用户，这两章的信息将填补他们在安全性知识方面的一些漏洞和缺口。

第三章涉及到控制用户对系统的访问。如果系统管理员不能驱除系统中的非法用户，那么在这场攻击与反攻击的斗争中就已失败了一半。目前，口令文件的弱点以及破解口令的方法几乎是众所周知的。在这本书里，系统管理员将学会有效的对策来控制 login 注册进程，并防止系统中出现不可靠的口令。系统管理员还将掌握有关正确设置系统文件的属主、权限的方法，并了解到系统文件中的问题是如何被有所企图的人利用的。本章同时还提供了一系列的 shell 程序，这些程序在正确地设置了属主、权限的基础上可以帮助系统管理员维护系统的安全性。

第四章主要涉及通讯与网络的安全性问题。通讯中的调制解调器 (后面称之为 modem) 及网络将会给系统的防御工作带来困难。由于有了远程访问，系统已超出了可以控制并监视的物理范围。系统管理员可以采用某些技术使得 modem 更加安全并改进

UUCP 系统的安全性。在这一章中还解释了 Internet 网的基本安全性，即哪些文件允许无口令注册，哪些文件可为安全的连接以及 NFS 的构造提供服务，如何正确地设置匿名的 ftp。这一章最后部分涉及到所谓的“受限环境”（这是一个可选项），即如何设置受限环境，每个环境到底能有多少安全性保障。

第五章探讨有关监视及检测任何可疑活动的技术。这章内容主要集中在如何判断系统是否受到攻击，如果受到攻击，应该采取什么对策等。同样，如果系统管理员已找到系统曾受到攻击的证据，本章将介绍如何追查滥用系统者。这一章同时还介绍了如何时刻监视系统的方法，并介绍了某些系统管理员成功地发现系统受到攻击的案例。最后，系统管理员还将了解到收集有关证据的原则。这些证据在对那些非法侵入系统、滥用系统的人进行起诉时会很有用的。

第六章论述了 UNIX 系统安全性的未来发展。目前存在有几种安全的 UNIX 版本，它们的安全特性将经受考验。改进的口令保护机制、强制性的存取控制，存取控制列表、安全注册、以及审计等特点构成了增强安全性的 UNIX 系统的基础。需要告诫的是：如果用户没有理解和遵循前几章所讨论的安全性问题及原则，那么即使 UNIX 版本是一个安全版本，用户在使用中也会使系统变得不安全。

1.3 计算机安全性

计算机的安全防卫并不包括卫兵、报警系统、红外探测器等物理防护措施。计算机安全性关心的是保护计算机内的信息以及控制对计算机资源的访问。这与物理防卫有着共同的目的，即任何时候都要对有价值的资源进行保护以防止偷窃、蓄意破坏及意外损坏。

详细地讲，计算机安全性包括三个不同的领域：

- 可记账性：通过确认用户以及记录下用户运行程序所做的操作，就可以“发现谁在干什么”，已记录下的这些操作可以帮助查出哪些活动比较可疑或断定哪些人破坏了系统的安全性；
- 系统保护：阻止任何用户冻结系统资源，如果某一用户长时间占用过多的系统资源而不释放（即冻结系统资源），则会影响系统对其他用户的服务，除此之外，系统保护还要防止对计算机资源的非法使用；
- 数据保护：控制对数据进行非法的读、复制、修改或删除。

在计算机安全性范围里，只有计算机和它的软件才是一切操作的“见证人”。为了使记账工作能正常进行，每个用户必须被唯一地标识。注册后，这个唯一的标识随着用户名一起用来标识记账记录、新产生的文件和目录的属主，并以此来判断某用户是否对某个文件及目录具有存取权限。

系统保护有两个针对的重点。将非授权用户拒之门外是系统保护针对的第一个重点。如果侵入者只能看到注册的提示符(login:)而无法进入系统，那么系统就不会丢失任何信息。系统保护针对的另一个重点经常被称之为“拒绝服务”。拒绝服务是指一个用户滥用系统，占据比他应得份额还多的资源，造成系统“拒绝”对其他用户进行服务，例如一次占据大量的磁盘空间或一次运行大量程序。在一次“拒绝服务”的攻击中，系统可能会由于攻击陷入困境，从而不能完成真正需要做的工作，甚至不能判断这是不是一次攻击。

数据保护控制着系统中用户对文件、目录的访问关系。系统必须根据用户的标识和文件的存取权限来决定这个用户是否可对某个文件进行读、写和执行操作。UNIX 文件的存取权限方案显得比较简单，但通过几种组合可构成多种变化，就象国际象棋一样，下棋规则虽简单，但下棋的招术千变万化。在人们了解了围绕着存取权限方案所做的攻击和破坏的原理后，人们就会对权限的设置格外谨慎，从而使 UNIX 系统的存取权限在某种程度上很好地起到了保护系统作用。

1.3.1 必要的安全性

在增加系统安全性的项目开始之前，应估算一下这一项目所需的经费。这里有一条简单的规则以决定是否有必要实施这一项目，即增加安全性所需的经费应低于由于安全性问题所造成的损失。在估算之前必须搞清楚哪些东西是有价值的，需要保护的。一串发射核武器的指令码是无法估价的，而且是至关重要的；而有关未来产品的市场的数据是比较容易估价的；私人的病理及心理信息是有价值的，如果这些信息泄露出来会造成个人名誉的损失；很明显，监视股票价格的计算机的平稳运行是有价值的，因为如果耽误几分钟就可能造成数以百万元的损失。

在估算增加系统安全性的同时，还必须考虑问题的另一面，即易用性。UNIX 系统之所以流行是因为它使得在同一计算机上的用户之间、同样运行 UNIX 系统的不同计算机之间，甚至在运行 UNIX 系统的计算机与运行完全不同的操作系统的计算机之间很容易进行数据交换。增加安全性不可避免地会带来副作用，即降低了交换数据的简易性，而这一点对 UNIX 系统的用户来说是非常重要的。如果增加安全性没有非常的必要，则会遭到用户的反对，甚至会出于逆反心理跟系统安全保护措施对着干，或绕过增加的安全性措施。如果已增加的安全性使计算机不易使用，则无人会去用它。易用性和安全性必须有所平衡。

在所有的 UNIX 用户中，程序员对安全性所带来的副作用最为反感。程序员花很长的时间编制他们的软件，如果幸运的话，这些软件在市场上销售出去，从而给他们编制下一个软件给予财政支持。虽然同行竞争者可以通过网络轻而易举地窃取别人的编程思想，程序员们仍希望网络不要被安全性措施捆住手脚。一条“Permission denied”（权限不允许）信息经常会引发程序员对保护文件的人一片愤恨。更糟的是，正是这些对安全性怀有抵触情绪的人负责设置发往用户的操作系统以及他们编写的软件的缺省安全性。值得庆幸的是，由于系统非法侵入者的存在，程序员对安全性的意识已大大增强了。

1.3.2 计算机安全性与道德行为

遵守计算机的操作规程是一个道德问题。未征得允许而进入或使用一台计算机就象闯入他人的办公室或私宅，把窗口开着决不意味着欢迎别人可以从窗而入、四处窥探。同样，未征得允许就不要进入他人的目录。注册目录就象是一个抽屉，之所以把东西保存在抽屉里是因为它们不宜放在外面让人一览无余。用户可以想象一下，当你走进办公室时，发现别人在搜劫你的抽屉，你的感觉如何？

如果用户没有来得及学会如何保护系统和数据的话，本书的某些内容可能会作为某些人闯入计算机系统的工具。本书并不专门介绍如何去闯入一个安全的 UNIX 系统，但显

然有些内容可被用来闯入一台防卫不好的系统。UNIX 系统的使用者在进行操作之前，必须认识到一台防卫不好的系统所隐含的危险性。用户可能会感到这种劝告很含糊，但读了本书的内容后，自然就明白这句话的含意了。

本书不是在鼓励任何人利用本书的内容进行计算机犯罪活动。那些由于中学生侵入计算机系统而使人们对计算机一片嘲讽的时代已经过去了。破坏计算机系统的行为已被视为一项严重的犯罪，不论是出于要浏览文件的目的，还是出于在未授权的情况下进入计算机系统的目的。本书中将讲到如何抓住有非法行为的用户，甚至是通过 modem 进来的用户，以及如何收集证据以进行犯罪起诉。

今天，计算机的"地下活动"很猖厥。对于一些人来说，侵入计算机是一种游戏，这种活动的犯罪性使这些人感到兴奋。这些人中有一些是程序员，其实这些程序员的自身的生计本来就依赖于是否能把他们的源程序保护好，但这些人对侵入别人的系统却很有兴趣。而另一些人则把计算机犯罪看成是赚钱的机会，他们把非法窃得的信息卖给某些组织。

在本书的读者面前有两种选择。如果读者想成为一个"守法者"，那么要特别注意本书所讨论的大部分内容是某些人成功地攻破 UNIX 系统安全性的案例，这方面读者须有所防备。如果读者想成为一个"违法者"，则要看是出于什么动机。如果只是出于发现系统安全性的漏洞而进行穿透试验从而改进系统，还是应受到鼓励的。有些公司奖励这些发现和报告系统漏洞的人员，因为他们在某种意义上为提高这些公司的产品性能做出了贡献。

1.3.3 计算机安全性的一个模型

计算机安全性相对来说是一个较新的概念。最早进行计算机安全性研究是在 60 年代末和 70 年代初。研究者们需要开拓有关安全性的新思路。在那时的安全性模型中，你需要保护的是纸上、胶片上或磁带上的信息，人们可以不用仪器或只用较简单的仪器就可看到这些信息。为了使你的机密安全、不被泄漏，你可使用物理安全防卫方法，例如把机密放在一间有人看守的房间里的保险柜中。对计算机安全性而言，你所保护的是电子编码信息，这些信息如果不利用计算机是无法看见的。系统管理员当然可以利用物理保护的方法来控制他人对计算机的接触。但系统管理员又怎么控制那些可以使用计算机的人呢？房间里上锁的保险柜对应于计算机内部又是什么呢？

计算机安全性所使用的许多方法与军队中所采纳的对信息的保护方法相类同，这一点并不奇怪。这种相似性可帮助读者建立一个易懂的模型，从而理解计算机的安全性模型。在本书的后面还要讲到这个比喻，所以这里用一点篇幅来讲述一个对虚构的 CIA 成员(名叫 Jack) 的观察，并与计算机安全性作对比说明。

早晨 8 点，Jack 驾车驶向 Virginia 北部 CIA 总部的停车处。卫兵在放 Jack 进入停车处以前，核对了 Jack 的身份证，并比较了身份证上的照片和 Jack 的脸。进大楼时的情况也是如此，Jack 停在玻璃门前，递上他的身份证作检查。在验证了身份证的照片和 Jack 的脸一致后，卫兵按了一个按钮，打开了门上的锁。事实上，Jack 认识这个卫兵，Jack 在走向办公室前，他们互道问候。当玻璃门在 Jack 身后关闭后，卫兵就在登记本上记录下 Jack 到达的时间。

同一天的晚些时候，Jack 需要检查几幅被列为机密的照片。Jack 走向照片管理员，向他解释要看照片的理由。管理员复查了一下被允许看照片的人员名单，找到 Jack 的名

字，并叫 Jack 在本上登记，然后从保险柜中拿出照片。在检查照片时，管理人员和 Jack 是在一起的，在归还照片后，管理员在本上记录下他在场的证明，并把这些东西放回保险柜中。傍晚，Jack 离开了那幢他工作的大楼，卫兵（这次可能换了一个）记录下他离开的时间。

请注意，所有计算机安全性所涉及的内容都在这个故事中提到了。在 Jack 进入停车处前，他必须递交身份证，以证明他是允许进入停车处的，即系统保护。卫兵比较了身份证上的照片与 Jack 的脸。身份证提供了身份，而照片提供了鉴证，这两项合起来证实了这个人和身份证上的人相对应。在前门，又进行了一次鉴证。卫兵本人是可以认出 Jack 是不是身份证上的那个人，而计算机却无此本领。

当 Jack 去看照片时，他遇到了存取控制机制，即数据保护。在他可以看到这些照片之前，管理员要检查一下他看照片的权利。当管理员在记录本上鉴字时，他就要对此行为负责。管理员是这件事的见证人，同样他也要负责把照片归还到保险柜中。

对计算机来讲，这些事情做起来大不一样。这次读者随着 Jill，进入 UNIX 系统进行工作，以考察计算机的有关工作过程。Jill 是在家里工作的，所以她命令她的 modem 拨入到 UNIX 系统中。当系统出现"login:"提示时，Jill 就键入她的名字（身份）。在回答"Password"提示时，Jill 键入她保密的口令（确认）。虽然这些步骤和前面故事中步骤的相似，但读者很快就能发现不同之处：任何人都可用 Jill 的用户名及口令进入系统。

当 Jill 注册时，有关信息作为数据存入计算机中。当她工作时，有关她活动的更多的信息也存贮了起来。Jill 和 Jack 不同的是，Jack 在楼内活动时须出示他的身份证，而 Jill 的标识从她进入系统后就一直伴随着她，作为 Jill 的一部分工作，Jill 必须检查某些敏感的信息。为做到这一点，她用了一个特殊命令，然后输入另一口令。她输入口令的成功或失败都将被记录下来。这样，她就可以存取那些敏感信息了。一旦结束这项工作，她又恢复到普通用户的身份。在结束一天的工作后，她退出系统，这时在 UNIX 系统中又建立了一个退出系统的记录。

1.3.4 可靠计算基

验明计算机用户、记录下用户的活动、控制文件的存取，这些任务都落在计算机的硬件和软件身上。计算机硬件包括：物理电子线路，处理器，总线及 I／O 设备，正是这些硬件支持了软件的运行。硬件的可靠性包括存储器的分区机制，即保护在一个分区内运行的软件不与在另一分区运行的软件和数据发生关系。操作系统是唯一可靠的软件。负责实施计算机安全性策略的硬件与操作系统软件的组合叫做可靠的计算基。(Trusted Computing Base)

现在，暂时忽略硬件在内存保护方面所起的作用，而将注意力集中到操作系统和几个程序上。CIA 例子中的卫兵和管理员的工作在操作系统里是由一些程序来完成的。例如，login 注册程序把用户键入的注册名作为一个参量，然后请求用户键入其口令，根据注册名和口令，login 程序可验明用户的身份（即用户所输入的口令必须与该用户在口令文件／etc／passwd 中的口令相符）。也就是说，这里的 login 程序起到了前门卫兵的作用。

现在定义两个术语，这两个术语在描述计算机安全性中经常引用到，即主体与客体。

主体是活动的实体，如人或以某个用户身份运行的程序。在 UNIX 系统中，活动的程序称之为进程，所以进程是主体。客体是被动的包含信息的实体，如文件、设备和系统本身。如果用户用 vi 读一个文件，则 vi 进程是存取文件的主体，而文件是客体。

基准监视器软件（reference monitor software）负责协调主体和客体的关系。基准监视器与 UNIX 系统核心不大一样。基准监视器只是程序中的一小部分，它负责存取控制。基准监视器必须很小以便于分析。如果基准监视器很大也很复杂，则很难通过检查代码来判断它工作得好坏。所以基准监视器是程序中被赋予存取控制这个特殊任务的一小部分。

基准监视器还使用了另外两个"资源"，一个是控制数据库，另一个是审计文件。控制数据库包含有关由主体存取的客体及其存取方式的信息。基准监视器用审计文件来记录它的活动。审计的工作范围很广，它可从不作任何记录开始到记录改变时间标记、记录哪个主体请求存取哪个客体、什么时间发生的请求，此请求是否被允许等。主体、客体、基准监视器、控制数据库以及审计文件之间的相互关系见图 1-1。

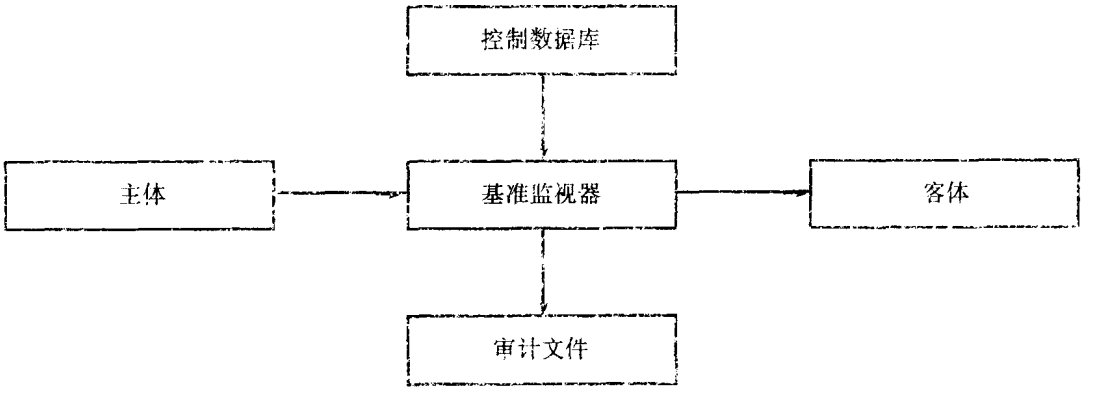

图 1-1　基准监视器负责连接主体与客体并将结果记录在审计文件中

UNIX 系统核心的某些部分是作为基准监视器来进行工作的。例如，Jay 用户想看一下 / etc / passwd 文件中的内容。Jay 启动 more 进程，并把客体的名字 / etc / passwd 作为参量。主体，即 more 进程，调用一个为读而打开 / etc / passwd 文件的系统调用。核心中的存取控制代码，即基准监视器之一，就检查 / etc / passwd 文件的允许权限（控制数据库）以决定 more 进程是否具有读权限，如果具有读权限，则把打开文件的信息返回给 more 进程。由于对 / etc / passwd 文件进行了读操作从而更新了文件的存取时间，这些变化都由基准监视器记录在审计文件中，见图 1-2 所示。

1.4 橘 皮 书

美国政府是最大的需要具有安全性的计算机系统的用户。政府中对安全敏感的机构必须遵守某些条例。这些条例是议会以法律的形式制定的。这些条例规定了如何处理敏感的信息及资料。通常这些资料是文件，但也可能是胶片、磁带或其他证据。保护敏感资料的

规程包括物理防卫（如卫兵、上锁的门、报警系统、保险柜）和安全分级系统。

安全分级系统定义了安全的等级及类别。安全等级有读者熟悉的一些名称，如绝密、机密、秘密和不保密。阅读保密资料需遵循一个简单的、常识性的原则，即：如果某人可以阅读机密文件，则他同样可以阅读秘密文件，但不能阅读绝密文件。

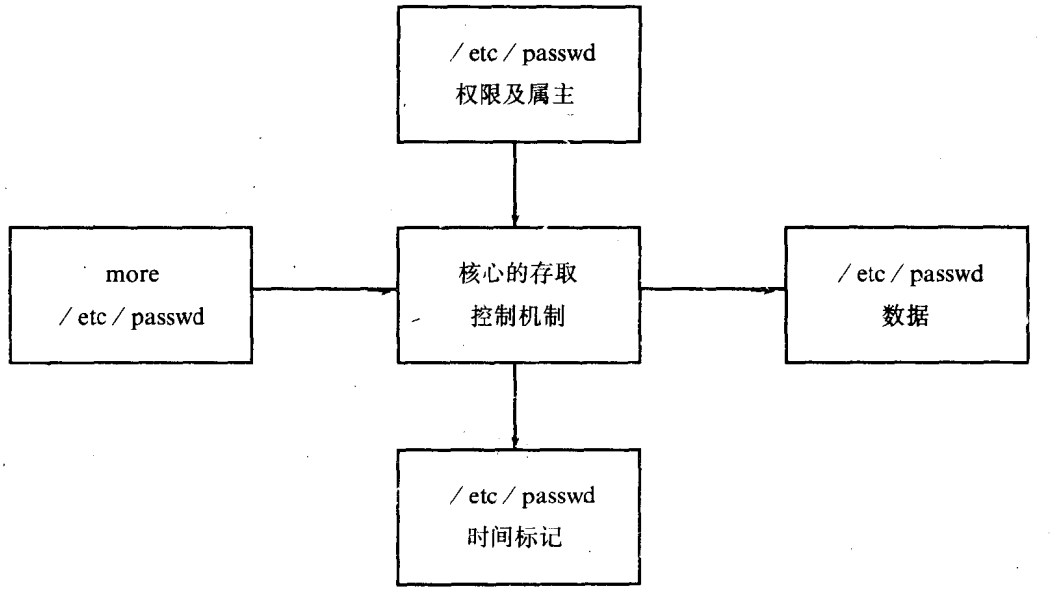

图 1-2　基准监视器在读取 / etc / passwd 文件的过程中所起的作用

安全类别与安全级别不大一样，但都比较易于理解。某个类别包含与一个主题有关的信息。作为一个分类的例子，这里给出了三个类别：中东、海军舰船、中美州。Oliver North 是为国家安全机构工作的，并负责处理向尼加拉瓜反政府武装筹集资金以及中东人质问题。他的活动涉及到我们上面所举的两个类别：中东及中美州，但不涉及到第三个类别：海军舰船。

分类的思想符合另一个安全原则，这一原则也是人们熟知的常识，即对某一秘密，知道的人越多就越有可能泄密。Oliver North 的工作属于中东类，而对于其他与其任务无关的文件他也无权过问。

国家计算机安全中心(NCSC)所做的工作就是定义判断准则，即判断计算机以及其软件与现有书面安全条例的符合程度。书面安全条例易于理解并且具有完善的规程。而计算机系统是非常复杂的。更糟的是，设计软件的程序员们与建立系统安全性之间存在着某种敌意。有些程序员认为信息应是无偿共享的，或者认为他们对自己设计的系统应享有存取特权。所以让程序员负责管理系统安全性犹如让窃羊贼牧羊。

为此，NCSC 设计了一套文档，名叫《可靠计算机评价准则》，以提供划分计算机系统安全级别的基准。(书的作者为了引起读者的注意，给这本书包上了橘色封皮，故又称这本书为橘皮书)。橘皮书于 1985 年由国防部出版，现已成为计算机系统安全级别的划分标准。虽然这本橘皮书并不是一本设计说明书，但现在设计者已把橘皮书中的思想溶于安全操作系统的设计中了。

1.4.1 橘皮书的定义

橘皮书并不是一本茶余饭后的消遣书，不懂计算机的人很难读懂。这里就它的主要内容作一些通俗的解释。橘皮书就以下几方面制定了准则：

▲一个清晰、严谨、文档齐全的安全性策略；

▲正确验证所有用户以及审计与安全性有关事件的能力；

▲确保系统正确地完成其可靠的操作；

▲为正确地使用和维护系统而提供的管理员手册及用户手册。

按照 NCSC 的准则来测试系统的安全性主要包括硬件部分和软件部分。整个测试过程对生产厂商来说是很昂贵的，而且往往需几年才能完成。一个申请某个安全级别的系统只有在符合所有的要求后才发给证书。

橘皮书做了一个有趣的假设：在一个可靠的系统中可能包含有不可靠的软件和蓄意破坏的用户。如果系统是可靠的话，那么应用软件就不必一定要在安全的环境中开发。可靠系统还必须控制用户的活动，以防他们有意或无意读写那些他们没有读写权限的数据。

橘皮书将重点集中在防止对敏感信息的损害方面，这一点很像它所模仿的书面安全条例，而忽略了防范象拒绝服务之类的对计算机安全性的攻击。拒绝服务是由于用户出于某种恶意或无意过多地占据系统资源的缘故。Morris Worm（见附录 G）就是一种拒绝服务式攻击，它让系统过载而不能正常工作。拒绝服务的攻击可使计算机在关键时刻瘫痪。

橘皮书依照了 Bell-LaPadula 的计算机安全模型，这个模型和书面安全条例很接近。Bell-Lapadula 模型定义了主体与客体的关系。这种关系是基于书面安全条例的分级与分类思想，并引进了一个新的术语：即支配。如果一个主体的级别大于或等于客体的级别，而且主体的类别包含了客体所有从属的类别，则这个主体对客体具有支配权。例如，如果用户的 vi 进程在中东类别中的机密级上工作，那么此用户就可以读中东类别中的机密级或更低级的文件。

在 Bell-LaPadula 模型中有两条基本原则，一条涉及读操作，另一条涉及写操作。对读操作来说，主体必须对客体有支配权，这与上面的例子是一致的。这种原则叫向下读原则，即你可以读与你同级或下级的客体。向下读被认为是简单安全原则。对写操作来说，客体必须对主体具有支配权。例如，当用户的 vi 进程在绝密级下操作或在不同的类别中工作时，用户就不能写一个机密级的文件。这种原则叫向上写原则。向上写原则防止了用户有意或无意把高密级信息写入低密级文件中，从而造成机密的泄露。向上写原则也叫作星号原则，因为 Bell-LaPadula 在书面上忽视了对这一原则的命名，只留下一个星号以提醒他们日后将名称补上。

橘皮书中还大量涉及了有关“保证”的问题。“保证”就是证明可靠系统的各个部分总是按照指定的方式工作。当安全等级提高时，“保证”就变得越来越复杂，所以做一个低级别的可靠系统比较简单些。

橘皮书并不完全遵循所有的军队安全条例。例如，某些资料只能在两个或两个以上的人在场时才能阅读。阅读资料的多重见证叫做双重管理，而在计算机上实现双重管理的工具还未开发出来。另外，在军队的安全条例中，可能会对一个文件的某一部分进行降级处理，以使一个文件可以存在于多个安全级别中，但橘皮书及 Bell-LaPadula 模型无法处理