装备科技译著出版基金

张涛 王金双 赵敏 译

信息安全实用教程

Applied Information Security

[瑞士] 大卫·贝森（David Basin）
帕特里克·沙勒（Patrick Schaller） 著
迈克尔·施莱普菲儿（Michael Schläpfer）

国防工业出版社
National Defense Industry Press

Springer

装备科技译著出版基金

信息安全实用教程

Applied Information Security

[瑞士] 大卫·贝森(David Basin)
帕特里克·沙勒(Patrick Schaller)　　著
迈克尔·施莱普菲儿(Michael Schläpfer)

张涛　王金双　赵敏　译

国防工业出版社
·北京·

著作权合同登记　图字:军-2014-131号

图书在版编目(CIP)数据

信息安全实用教程/(瑞士)贝森(Basin,D.),(瑞士)沙勒(Schaller,P.),(瑞士)施莱普菲儿著;张涛,王金双,赵敏译.—北京:国防工业出版社,2015.11
书名原文:Applied Information Security
ISBN 978-7-118-10348-9

Ⅰ.①信...　Ⅱ.①贝...②沙...③施...④张...⑤王...⑥赵...
Ⅲ.①信息安全—高等学校—教材　Ⅳ.①TP309

中国版本图书馆CIP数据核字(2015)第275060号

※

国防工业出版社出版发行
(北京市海淀区紫竹院南路23号　邮政编码100048)
北京嘉恒彩色印刷有限责任公司
新华书店经售

*

开本710×1000　1/16　印张11¼　字数200千字
2015年11月第1版第1次印刷　印数1—2000册　定价68.00元

(本书如有印装错误,我社负责调换)

国防书店:(010)88540777　　发行邮购:(010)88540776
发行传真:(010)88540755　　发行业务:(010)88540717

感谢家人的支持，感谢瑞士苏黎世联邦理工学院
信息安全研究所成员的投入和反馈

译者序

市场上关于信息安全的书籍种类繁多，内容各有侧重。本书着眼于理论与实践相结合，内容组织上始终贯穿着“知行合一”的教育理念。为了实践这一理念，原书作者还构建了信息安全攻击与防护的典型虚拟机环境（已随书提供）。该环境的典型性和实用性已由作者在瑞士苏黎世联邦理工学院的多年教学实践中得到了验证。

译者从事信息安全专业一线教学多年，迫切感觉到实践能力培养的重要性，并欣喜地发现本书恰好符合我们的期待。通过引入本书的部分实验，我们发现，实际动手操作能够促进学生将理论知识与实践认识统一起来，从而达到较好的教学效果。

正如原书作者所说，本书适用于有一定信息安全理论基础的学生，尤其适合在理论课之后的自学。

感谢潘林老师和杨海民、苏煜、鲁小杰等研究生，他们参与了本书译稿的校对工作，并对部分实验进行了再次验证。

虽然翻译工作很辛苦，但一想到这本书也许能给同行老师和有兴趣的同学带来一些帮助，我们就感到信心百倍。由于水平所限，书中不当之处在所难免，欢迎读者们批评指正！

译 者

2015 年 12 月于南京

前　言

在过去几十年中，信息安全已经从要由军事密码学家研究的专业性主题转变成一般性主题，与每一个希望更好地理解、发展或使用现代信息和通信系统的专业人士息息相关。大多数信息安全课程强调理论和基本概念：密码学、算法、协议、模型和特定的应用程序。从为阅读者提供对该主题的基本了解的角度来说，这是至关重要的。但是信息安全最终需要身体力行，并把这些想法付诸于工作实践。这就是本书的切入点。

编制本书的主要目的是为了提供动手实验的指导，以辅助偏理论性的教材。我们从实验室的角度来学习信息安全，学生们在实验室进行实验，就像他们在其他课程（比如物理化学课）上做的实验一样。我们的目的是让学生通过把课上所学的理论知识直接付诸实践，看到第一手的实践结果及其奥妙之处，从而能更好地理解这些理论。就像其他实验课程一样，本书并不是想要代替理论课及其相关教材，而是作为一种补充，来夯实和拓展学生的知识。

本书源自于瑞士苏黎世联邦理工学院开设的一门实验课程，这门课程始于2003 年，至今仍在开设。其课程宗旨正如前面提到的：提供一个与校内有关信息安全领域的偏理论性课程相适应的实践机会。选修这门课程的学生都会得到这本书，里面包含三个网络化的虚拟机，而这些虚拟机是在虚拟环境下运行的。这本书使读者面对不同信息安全领域主题的各类问题，而这个软件则允许学生进行实验，以便把先前学到的理论概念能够应用到实践。

本书主要侧重在网络的安全、操作系统及 Web 应用程序上。以上每一主题涉及的内容都很广泛，甚至可以独立成书。我们把注意力放在这些领域的核心安全问题上，比如，身份验证和访问控制、日志记录、典型的网络应用程序漏洞及证书等一些确立已久的主题。这非常符合我们的意图——用在实验室中的实践来弥补偏重理论的信息安全课程的不足。

本书涉及的软件已在网上发布，可从 www. appliedinfsec. ch 下载。通过使用虚拟化技术，课程所需要的软件均已完全包含在虚拟机中。该软件可在大多数操作系统中运行，包括 Windows、Linux、Macintosh 和 OpenSolaris，支持 VirtualBox 的，这是针对于 x86 和 AMD64/Intel64 平台的一个免费的虚拟化环境。

如何使用本书

本书可用于以下两种方式。

第一，可用于自学。当我们在瑞士苏黎世联邦理工学院教授本门课程时，学生独立学习了所有章节并回答了所给问题。学过信息安全基础课程且有 UNIX 衍生系统操作经验的学生能够独立完成大部分的练习。为便于自学，本书附录附有答案。

第二，本书可以用于大学的实验课或在行业内使用。就拿我们在瑞士苏黎世联邦理工学院开设的课程来说，我们在实验中补充了一个项目。在该项目中，学生会进行分组，且每组最多不超过 4 人。他们的任务是依照规定开发出一个完善的系统。这个系统必须以一种运行在 VirtualBox 虚拟机上的方式被提交出去。到课程结束时，虚拟机会分配到个组，并且每一个系统会由一个不同的小组来进行检测。这门课程的总分会基于项目得分和期末考试得分来综合计算。

不论采用哪种方法学习，最好按照已给的顺序阅读各章节。第 1 章提供了基本安全原则的背景知识，这些知识贯穿于整本书。第 2 章介绍了 VirtualBox 环境，做练习会需要相关知识。接下来是两个比较独立的部分：第 3 章到第 5 章讲的是有关网络和操作系统安全的知识，第 6 章到第 7 章讲的是 Web 应用程序安全和证书。但是，会有一些重叠和交叉部分，如应用程序会使用到网络服务，并且会在操作系统上运行，因此我们建议按顺序重复学习这些交叉部分。

本书的最后一章讲述风险分析。这一章与其他各章均不同，别有特色。它详述了分析系统整体安全，通用程序，也就是说，分析整体而不是仅仅分析每个部分。若是自学，这部分则可省略。但对于那些开展此项课题的读者来说，这一章也是至关重要的，并且，就其本身而言，这一章也是一个重要的主题。

本书有四个附录。附录 A 和附录 B 给出了可行项目详细示例，这已在瑞士苏黎世联邦理工学院成功使用。附录 C 提供了 Linux 的简要概述和各种对练习及课题有用的实用程序。附录中涉及的都是一些基本的资料，根据我们的经验，对那些先前经验有限的读者来说，这些资料还是有用的。附录 D 中提供了本书中所有问题的答案。

符号和术语

本书使用的符号和术语大多依照惯例。首先，与常见的安全书籍一样，本书故事里的人物名字都是 Alice、Bob 和 Mallet。我们选取这些名字表明不同的目的，相同名字的不同字体也代表着各自特定的意义。Alice 和 Bob 是忠诚的代理；Mallet 则是一个恶意代理，会入侵系统，在一定程度上危害系统的安全。每个代理都拥有自己指定的虚拟机，并且虚拟机名字与代理名字相同。`alice` 主机使用图形用户界面运行桌面操作系统，`bob` 主机则被配置为只提供访问命令行的服务器。代理 Mallet 的虚拟机 `mallet` 运行的桌面操作系统能够提供侵入其他系统的工具。另外，代理的名字还可以表示某些应用的用户名，如 *bob* 表示 Bob 在 bob 主机的登录名。

我们经常需要描述系统的输入和输出，我们会使用打字机字体来表示指令、

命令行输入、输出和文件名。

本书提到的软件是基于 Linux 的。我们阐述的所有理念适用于类 UNIX 系统,如 BSD、Solaris、Mac OS 等。大部分指令在这些类 UNIX 系统也能工作,可能出现少许变化。一般我们用术语 Linux 指代任何类 UNIX 系统。

本书也含问题和练习,它们的区别如下:问题插入在本书的讲解中,读者阅读时能够检查学到的知识,参考答案在附录 D,我们使用以下方式来突出问题。

问题 0.1 这个问题的答案在附录 D。

大部分章节末尾含有练习,这些练习以问题形式出现。练习是课程的一部分,教师可能会将它用于作业或考试,从而检查学生的知识。因此本书不提供这些练习的答案。

最后,我们用颜色突出的方框显示重要的文本。我们将▷符号放在我们希望读者进行操作的内容前,并用颜色突出显示这些内容。

▷ 这是读者应该操作的任务。

我们也用高亮颜色显示原则和设置等。

不可以关闭、窃取应用程序或使其无效。

简便起见,对明显的和不太重要的控制台输出使用简写表示。比如,我们会省略当前的工作目录,使用 alice@ alice: $,而不使用 alice@ alice:/var/log $ 。

历史和感谢

瑞士苏黎世联邦理工学院的信息安全实验室是在 2003 年由 David Basin 和 Michael Näf 创建的。他们在以虚拟环境为基础的信息安全方面设计了一系列实验,写了一个以实验为基础的脚本,并将其运用到他们开设的实验室课程当中,2003—2004 年冬季学期开始该课程。由于其鲜明的实践方法,该课程持续受到高年级学生的欢迎。2007 年,当 Michael Näf 为建立 Doodle 公司而离开系里时,Patrick Schaller 接管了实验室,并和 David Basin 及助教一同授课。

相对于 2004—2009 年间的版本,本版本只做了稍许的修订,课程的教学材料,包括脚本和相关软件基本保持不变。本书多年来未做过较大变动,这就证明了本书从开始就一直保持着高质量,而且也说明了要改变已提供给学生的虚拟机着实困难。终于,在 2010 年,我们认为对脚本和软件进行重大修改的时机已经成熟了。我们沿用了脚本的初始结构,修订、更新、修改了相当部分的内容。另外,我们将虚拟机替换为了最新版本。以上就是这本书改版的原因。

许多人为这门课程提供了素材,其中许多素材以多种形式被收录进了本书。David Basin、David Gubler、Manuel Hilty、Tilman Koschnik、Michael Näf、Rico Pajarola、Patrick Schaller、Paul Sevinc 和 Florian Schütz,都是本书第一版的重要贡献者。其中 Michael Näf 更是推动了早些年课程素材的整理工作。David Basin、Luka Malisa、Pascal Sachs、Patrick Schaller 和 Michael Schläpfer 等为本书进行了第

一次重大修改。我们感谢本书所有的合作者的大力相助，使本书的出版成为可能。我们同时也感谢创作书内插图的 Barbara Geiser，以及帮助审稿的 Jeffrey Barnes。

David Basin
Patrick Schaller
Michael Schläpfer
2011 年 8 月于瑞士苏黎世

目　录

第1章

安全原则

本书的目标是帮助读者通过实验来提升对信息安全的理解。我们并不覆盖所有相关理论，而是假设读者已通过学习其他课程或书籍，具备了密码和信息安全的基本知识。然而，我们将概述一些与实验相关、更重要的核心理念。

第1章横向概括了后续章节相关原则，这12项安全原则提供了将安全融入系统设计的指南。这些原则的叙述将尽可能一般化，以便帮助读者发现后续章节更具体的设计实践中的共性。

1.1 目 标

阅读本章后，应理解12项原则，且能够将其解释给尚不熟悉该原则的IT专家。此外，应能够提供一些遵循上述原则示例或违反安全原则而导致的安全问题。

1.2 问题情境

本书专注于系统安全，典型的系统是指运行网络服务软件的计算机平台。安全通常是由描述授权行为的策略所定义的。例如，电子邮件服务中，应只允许注册该项服务的合法用户访问其邮箱。如果恶意、非法用户可访问合法用户的邮箱（例如，通过获取该用户的认证凭证），便违反了安全策略。

电子邮件服务的例子说明了信息安全的复杂性。即使一个提供电子邮件服务的简单平台也会包含Web前端、邮件服务器和数据库等多个子系统。因此，恶意用户可通过许多可能的方法攻击系统。除了获取有效用户名和密码组合，他还能利用它们所在的网络服务器、数据库、操作系统中的漏洞。一个漏洞便能破坏系统的安全性，这样便简化了攻击者的工作。相反地，负责系统安全的人员必须尽一切努力全盘考虑整个系统。安全不能孤立于某个单一系统组件，安全系统的建立是贯穿于系统开发、部署和运行全过程的结果。

下一节给出了直接或间接来源于科学工程实践[5,12,18,22,24]的一系列安全原则。这些原则为开发过程中如何考虑安全及如何评估已有的方案提供了指南。

1.3 原　则

传统的信息安全目标包括保密性、完整性、可用性和不可抵赖性。以下所选安全原则有助于实现安全目标和分析系统安全。本次选择不求全面，但却包含了与具体领域无关的最重要准则。也就是说，这些原则既适用于操作系统安全，也适用于应用和网络安全。

为扩大适用范围，我们按照主体和客体抽象地表述这些原则。主体是主动实体，如用户或代表用户行动的系统。客体是将信息作为数据存储的被动容器。访问某个客体通常意味着访问其包含的数据。客体包括记录、数据块、页面、区段、文件、目录和文件系统等。

谈到这些原则，我们经常会提到 Saltzer 和 Schroeder，他们关于主体的论文[18]是经典著作。尽管他们在 35 年前编写的这篇文章，但在今天看来，里面所陈述的大多数原则依然明确且适用。

1.3.1 简单性

保持简单化。

该原则适用于设计或实现系统所涉及的所有工程和实现任务。更一般地说，它是解决任何问题和反映解决方案质量的良好指南。解决方案越简单，越容易理解。

简单性是系统设计、开发、运行和维护及安全机制等所有方面的理想特性。和复杂系统相比，简单系统包含缺陷的可能性较小。此外，简单系统比较容易分析和审查，因此，更容易确立可信性。

Saltzer 和 Schroeder 称这种原则为机制的经济性。他们指出了该原则对于在软硬件层次上进行检查的保护机制的重要性。这类检查若想成功，小而简单是至关重要的。

1.3.2 开放式设计

系统安全性不应依赖于保护机制的保密性。

Saltzer 和 Schroeder 将该原则精确地描述为：

机制不应依赖于潜在攻击者的无知，而是依赖于特定的、更易受保护的密钥或密码。将保护机制与保护密钥分开，有利于多个审核人员进行检查，而不必担心审核本身会破坏防护措施。

该原则在密码学中被称作柯克霍夫斯原则[7]，它指出公开除了密钥以外的信息都不会影响密码系统的安全性。

保密并非易事，必须把秘密存储在大脑、内存、磁盘或外部设备等处并提供相应的保护。因此，需保密的信息总量应最小化。

例 1.1 我们不设计只有授权人士懂得开和关的门。相反，我们设计配有标准锁的标准门（具有不同保护级别）并依赖于对相关钥匙的保护。

1.3.3 分隔

把资源划分为具有同样需求的隔离组。

分隔是指把资源划分为相互分离的组（又称为分区或区域），除了一些受限或受控形式的信息交换外，每组可单独存在。分隔原则被运用于计算机科学的不同领域，如程序设计中的函数和变量被分组并放到独立的模块或类中。

例 1.2

（1）敏感应用通常运行在独立的计算机上以便将攻击的影响最小化。如果一台计算机及其应用程序被入侵，它将不会让其他应用也受到攻击。这对基于 Web 的应用的各层次也适用。通常这些应用放置在独立的服务器上，比如是为了防止应用服务器上商业逻辑被破坏而影响到数据库。

（2）可以采用其他方法分离应用程序和运行系统。基于软件方案可实现单一物理机器上的应用隔离，常用技术为 UNIX 系统中内核/用户模式、VirtualBox、VMware、Xen、VServer、Jail、chroot 作业系统或者是硬盘分区。

（3）分隔方法广泛应用于网络。常使用如防火墙之类的过滤装置，把网络分为单独区域，区域间通信受策略限制。其目标是增加不同区域内主机攻击机器或服务器的难度。常用的区域概念包括内网以及与外网和互联网有连接的非军事区（DMZ）。

（4）分隔方法在软件开发中同样重要，它得到了大部分现代程序语言支持。基于语言的机制（如封装性、模块化和面向对象）可以促进和实施分隔方法。这些机制同样有助于构建更安全的系统。

分隔方法有多项优点。第一，由于隔间包括相同（相似）需求的资源，所以分隔方法促进整项操作简单化。例如，相对于细粒度访问控制，粗粒度访问控制则可以较容易地理解、实现、评审和维护。第二，通常可以把攻击产生的问题、操作问题、意外事故出现的问题和类似问题分离为单独隔间。因此分隔方法减少了问题的负面影响，为处理问题提供了机制。例如，可能将受影响的网络部分与互联网断开连接，或停止不当行为过程。第三，在某种特殊隔间中可设置高安全敏感性功能，在此特殊隔间同样可加强安全措施和策略。

数据和代码分离是分隔原则在软件工程中的重要应用。许多协议和数据结构把数据和代码混合，这样可能会出现安全隐患。堆栈上的数据和代码混合可导致缓冲区溢出。Office 文件中的数据和代码混合可能会导致包含宏病毒的恶