安全技术经典译丛

PEARSON
Addison Wesley

ROOTKITS: SUBVERING THE WINDOWS KERNEL

ROOTKITS

——Windows内核的安全防护

剑

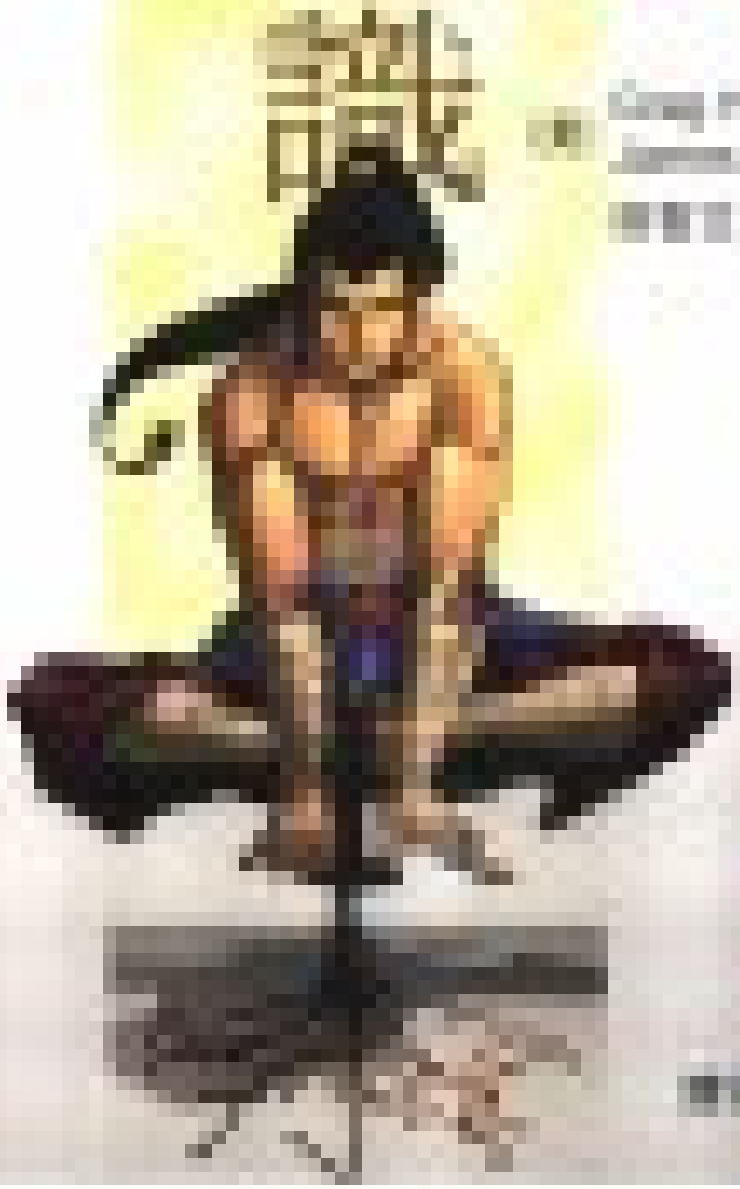
(美) Greg Hoglund
James Butler 著
韩智文 译

PEARSON
Education

清华大学出版社

ROOTKITS

——Windows内核安全编程



Rootkits

——Windows 内核的安全防护

(美) Greg Hoglund 著
James Butler
韩智文 译

清华大学出版社

北 京

北京市版权局著作权合同登记号 图字：01-2006-6351

图书在版编目(CIP)数据

Rootkits——Windows 内核的安全防护/(美)霍格兰德(Hoglund, G.)，(美)巴特勒(Butler, J.)著；韩智文译. —北京：清华大学出版社，2007.4
书名原文：Rootkits: Subverting the Windows Kernel
ISBN 978-7-302-14652-0
Ⅰ.R… Ⅱ.①霍… ②巴… ③韩… Ⅲ.窗口软件，Windows—安全技术 Ⅳ.TP316.7
中国版本图书馆 CIP 数据核字(2007)第 020714 号

责任编辑：曹 康 张 云
装帧设计：孔祥丰
责任校对：成凤进
责任印制：王秀菊
出版发行：清华大学出版社 **地 址**：北京清华大学学研大厦 A 座
http://www.tup.com.cn **邮 编**：100084
c-service@tup.tsinghua.edu.cn
社 总 机：010-62770175 **邮购热线**：010-62786544
投稿咨询：010-62772015 **客户服务**：010-62776969
印 刷 者：北京市清华园胶印厂
装 订 者：三河市兴旺装订有限公司
经 销：全国新华书店
开 本：185×230 **印 张**：19.5 **字 数**：371 千字
版 次：2007 年 4 月第 1 版 **印 次**：2007 年 4 月第1 次印刷
印 数：1～4000
定 价：39.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题，请与清华大学出版社出版部联系调换。联系电话：(010)62770177 转 3103 产品编号：022322-01

专家推荐

rootkit 不仅是最新的，它还定义了什么是最新的。它处于真正的前沿。作为该主题唯一的著述，《Rootkits——Windows 内核的安全防护》将是所有 Windows 系统安全研究人员或安全编程人员都感兴趣的图书。本书阐述具体、研究深入、技术资料翔实，其技术细节、研究水平，以及开发相关示例所投入的时间都令人印象深刻。一言以蔽之，这是一部杰作。

——Tony Bautts，安全顾问，Xtivix 公司 CEO

对于负责 Windows 系统安全的任何人来说，本书都是必读的。安全专家、Windows 系统管理员以及程序员通常都希望理解 rootkit 作者所用的技术。当许多 IT 和安全专家仍在担心最新的电子邮件病毒或者如何完全安装本月的安全补丁之时，本书作者展示了对 Windows 操作系统最秘密最严重的一些威胁。只有理解了这些攻击技术，才能正确地防护所管理的网络和系统。

——Jennifer Kolde，安全顾问，作家，讲师

Greg 和 Jamie 无疑是在破坏 Windows API 和创建 rootkit 时所需要的专家。这两位大师合力揭开了笼罩在 rootkit 周围的神秘面纱，将其大白于世。所有对 Windows 系统安全感兴趣的人都应该将本书置于必读书目的前排。

——Harlan Carvey，*Windows Forensics and Incident Recovery* 一书作者

还有什么比被占有更为糟糕的？不知道。

请阅读 Hoglund 和 Butler 关于 rootkit 技术的第一流作品来弄明白被占有的含义。rootkit 处于恶意黑客工具集合(包括反编译器、反汇编器、错误注入引擎、内核调试器、有效负载收集工具、覆盖分析工具以及流量分析工具)中的颠峰。从《软件剖析——代码攻防之道》一书的结束处开始，本书展示了攻击者如何隐身于清晰视野之中。

rootkit 带来下一波攻击浪潮，它具有极为强大的能力。类似于其他类型的恶意代码，rootkit 靠潜行起家。它们对标准的系统监控器隐匿自身，利用钩子技术、trampoline 技术和补丁方法完成工作。复杂的 rootkit 在运行时能够使得一般的系统行为监视程序无法轻易地检测到它们。因此 rootkit 只向知道它在运行中并准备接受命令的知情者提供了内部访问。内核 rootkit 可以隐藏文件和运行的进程，从而提供了进入目标机器的后门。

理解终极攻击者的工具为系统防护人员提供了重要的操纵能力。没有人比 Hoglund 和 Butler 更适合向您传授关于 rootkit 技术的细致深入理解。最好占有本书，而不要被占有。

—— Gary McGraw，博士，Cigital 公司 CTO，《软件解析——代码攻防之道》和《构建安全的软件：避免产生软件安全问题的正确方法》(均由清华大学出版社引进并出版)的合著作者。

前　言

rootkit 是持久且无法察觉地存在于计算机之上的一组程序和代码。

历史背景

我们对 rootkit 产生兴趣来自于我们在计算机安全领域的本职工作，但对于该主题的探索迅速发展成为一种个人使命。这导致了 Hoglund 建立 rootkit.com 网站，一个致力于反向工程和 rootkit 开发的论坛。我们两人都深深投入到了 rootkit.com 中。Butler 首先通过该网站在线联系上了 Hoglund，因为 Butler 得到一个需要测试的强大的新 rootkit[1]，名为 FU。他向 Hoglund 发送了一些源代码和预编译的二进制代码，然而疏忽中却没有发送内核驱动程序的源代码。令 Butler 惊奇的是，Hoglund 成功地将预编译的 rootkit 加载到自己的工作站上，并报告说 FU 似乎运行良好。之后我们彼此的信任才不断加深[2]。

我们两人都长期受到一种几乎执拗的想法的驱使，即对 Windows 内核进行反向工程。就像当有人说我们无法做成某事时，我们就一定要完成它。了解所谓的计算机安全产品如何工作并发现它们的原理会令人感到非常满足。这必然导致更好的保护机制。

某种产品声称提供了某种级别的保护并不一定意味着它实际能够如此。在扮演攻击者的角色时，我们总是占据着优势。作为攻击者，我们只需想到防御者没有考虑的一件事即可。而另一方面，防御者必须考虑到攻击者能够完成的所有可能的事。这个数量对比有利于攻击者。

1　Butler 对 rootkit 的恶意用途不感兴趣。相反，他迷恋于内核修改所产生的作用。这导致他开发了最早的 rootkit 检测程序之一 VICE。

2　Hoglund 有时仍在怀疑，该原始版本的 FU 是否仍在他的工作站上运行。

几年前，我们联合开设了培训课程“rootkit 技术的攻击问题”。该课程最初只准备了一天的材料，现在已发展为数百页的笔记和示例代码。该课程的素材最终成为本书的基础。现在我们每年在 Black Hat 安全大会上以及私下场合多次提供 rootkit 培训课程。

在培训一段时间后，我们决定加强联系，现在共同就职于 HBGary 公司。在这里，我们每天都要处理非常复杂的 rootkit 问题。本书中，我们基于自身经验涵盖了当今 Windows 用户所面临的并且很可能在将来会继续增长的威胁。

读者对象

本书面向对计算机安全感兴趣并希望对安全威胁获得更真实感受的读者。关于入侵者如何获取计算机系统访问权的问题已有大量著作面世，但关于入侵者进行了这种初始访问之后会发生什么情况的问题还鲜有涉及。如书名所示，本书介绍了入侵者能够执行哪些工作来掩护其在被攻陷机器上的存在。

我们认为，包括微软公司在内的大多数软件供应商都没有严肃地对待 rootkit。这是我们出版本书的目的。本书中的素材对于多年从事于 rootkit 或操作系统领域的一些人员来说并不是爆炸性的——但对于大部分人来说，本书应该能够表明 rootkit 是一种严重的威胁。它应该证明了扫描器或桌面防火墙从未提供足够的保护；还能够证明 rootkit 可以闯入您的计算机并在其中驻留多年，而您对此却一无所知。

为了更好地传播 rootkit 知识，我们从攻击者角度撰写了本书的大部分内容，但以防御者的姿态结束本书内容。开始了解攻击者的目标和技术之时即是了解自己系统的脆弱点以及如何消除其缺陷之时。阅读本书有助于您改进自身系统的安全或者在购买安全软件时做出明智的决策。

预备知识

由于所有代码示例是使用 C 语言编写的，因此掌握基本的 C 语言概念(尤其是指针概念)有助于获得更深入的理解。若不具备编程知识，仍应该能够继续学习并理解所有威胁，但无需理解特定的实现细节。本书的某些内容依赖于 Windows 设备驱动程序体系结构的原理，但无需设备驱动程序的编写经验。我们将带领您编写第一个 Windows 设备驱动程序，由此继续学习。

本书适用范围

本书介绍 Windows 系统的 rootkit，但大多数概念也适用于其他操作系统，如 LINUX。重点是内核 rootkit，因为它们最难以检测。Windows 的许多公开 rootkit 都是用户空间的[3]，由于它们不必理解在文档中未说明的内核是如何工作的，因此最易于实现。

本书并不针对现实中的特定 rootkit，而是描述了所有 rootkit 使用的一般方法。每一章引入一种基本技术，解释其目的，并通过代码示例演示如何实现它。基于这些信息，应该能够以无尽的方式对示例进行扩展，以便执行各种任务。在内核中工作时，您实际上只受自己的想像力所囿。

本书中的大部分代码可以从 rootkit.com 网站下载。在整本书中，对每个示例都给出了具体的 URL。其他 rootkit 作者也在 rootkit.com 公布了有助于跟踪最新研究发现的成果。

3　用户空间 rootkit 不利用内核层次上的改动，而是只依赖于用户程序的改动。

目　录

1 销声匿迹

当前已有大量书籍讨论渗透计算机系统和软件的技术，涉及如何运行黑客脚本，编写缓冲区溢出的利用程序，以及构造 shell 代码等。比较有名的例子包括 *Exploiting Software*[1]，*The Shellcoder's Handbook*[2]和 *Hacking Exposed*[3]。

与上述著作不同，本书并不涉及攻击技术，而是研究攻击者在侵入系统后如何继续存留下去。除了计算机取证分析方面的书籍之外，鲜有著作讨论在成功渗透系统之后所进行的工作。取证分析领域的书籍只进行防御性的讨论——如何检测攻击者以及如何对恶意代码进行反向工程，而本书采取了攻击性的方法，其内容是关于在未检测到的情况下渗透计算机系统。毕竟若要使得渗透活动随时间流逝而能够证明是成功的，就不允许被检测到。

本章介绍 rootkit 技术及其一般性的工作原理。rootkit 只是计算机安全领域的组成部分之一，但它对于许多攻击的成功来说是关键的。

rootkit 自身并不具有恶意性质，然而恶意程序却可以利用它。理解这种技术对于防御当前的攻击活动是至关重要的。

1.1 攻击者的动机

计算机中的后门(back door)是一种获取访问能力的秘密方式。后门技术流行在许多好莱坞电影之中，表现为能够访问高安全性计算机系统的秘密口令或方法。但

1 G. Hoglund 和 G. McGraw，*Exploiting Software: How to Break Code*(Boston: Addison-Wesley, 2004)。参见 www.exploitingsoftware.com 网站。

2 J. Koziol, D. Litchfield, D. Aitel, C. Anley, S. Eren, N. Mehta 和 R. Hassell，*The Shellcoder's Handbook* (New York: John Wiley & Sons, 2004 年)。

3 S. McClure, J. Scambray 和 G. Kurtz，*Hacking Exposed*(New York: McGraw-Hill, 2003 年)。

后门并不只是活跃于银屏之上，它们是非常真实的存在，可用来窃取数据、监视用户以及发起深入计算机网络的攻击。

攻击者出于多种考虑可能会在计算机上留后门。突破计算机系统是一件困难的事情，因此攻击者一旦取得成功后，总希望保持已占据的领地，甚至希望使用被攻占的计算机对网络发起其他更深入的攻击。

攻击者渗透计算机的主要目的是收集情报。为此，他要能够监视击键动作，观察随时间变化的行为，嗅探网络上的报文，并从目标上偷取[4]数据。所有这些功能都需要建立后门，因为攻击者希望情报收集软件驻留在目标系统上运行。

另外，攻击者也会通过渗透到计算机中对其进行破坏，这时攻击者可能会在计算机上放置逻辑炸弹，将其设定为在特定时刻摧毁计算机。炸弹处于等待阶段时需要悄无声息地驻留。即使攻击者日后不再需要访问系统后门，但它还是一种软件遗留，仍必须是无法检测的。

1.1.1 潜行的角色

为了达到无法检测的目的，后门程序必须采用潜行技术。然而，大多数公开可用的“黑客”后门程序并不是非常秘密的，会出现许多错误情况。这主要是因为开发人员希望在后门程序中实现所有功能，包括尽人皆知的洗碗池，例如 Back Orifice 或 NetBus 程序。这些后门程序具有丰富的特性列表，其中一些甚至会愚蠢地弹出 CD-ROM 托盘。这种事对于办公室幽默来说是好笑的，但不该是专业攻击操作中使用的功能[5]。攻击者如果粗心的话，就可能会暴露自己在网络上的存在，导致整个攻击操作失败。基于该原因，专业攻击操作通常需要特定的自动化后门程序——只完成单项功能且绝不执行其他功能的程序。这样可以确保结果一致。

如果计算机操作员怀疑自己的机器或网络已遭到渗入，他们可以执行取证(forensic)发现[6]，查找不同寻常的活动或后门程序。反击取证分析技术的最佳方式是潜行技术：若没有怀疑到攻击存在，则不会对系统进行取证分析。攻击者可通过多种方式使用潜行技术。例如，将网络流量减至最小并避免在硬盘上存储文件，从而

4 偷取：从某个位置传出和删除；将数据副本从一个位置传输到另一个位置。

5 在这种情况下，专业性意味着由法律实施机构、监狱测试人员、红色团队或类似人员所执行的某种被核准的操作。

6 关于计算机取证分析的资料，参见 D. Farmer 和 W. Venema 的著作 *Forensic Discovery*(Boston: Addison-Wesley, 2004 年)。