

<<灰帽黑客>>

图书基本信息

书名：<<灰帽黑客>>

13位ISBN编号：9787302301509

10位ISBN编号：7302301506

出版时间：2012-11

出版时间：清华大学出版社

作者：[美]Allen Harper,[美]Shon Harris

页数：586

字数：895000

译者：杨明军,韩智文,程文俊

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## &lt;&lt;灰帽黑客&gt;&gt;

## 前言

我已经受够了一场战争，更别提再来一场了。

——托马斯·杰弗逊 我不知道第三次世界大战的武器会是什么样子。

但我知道第四次世界大战战场上用的肯定是棍棒与石头。

——阿尔伯特·爱因斯坦 兵法非常简单。

找出敌人哪里，尽快到达那里，尽可能凶狠地打击他，而且要挺住。

——尤利西斯·S·格兰特 本书的目标是帮助培养更多技术精湛的、专注于抵御恶意黑客攻击的安全专家。

事实一再证明，对敌方的了解是非常重要的，包括他们的策略、技能、工具和动机。

企业和国家所面临的敌手非常专注，而且技艺超群。

我们必须携起手来才能理解敌方的行动过程和流程，以确保我们能够正确地挫败敌方具有破坏性的恶意攻击。

本书的作者希望为读者提供他们认为这个行业所需的信息，即对负责的而且在意图和物质方面真正合乎道德标准的正义黑客技术的整体性讨论。

这也是为什么本书一开始就给出正义黑客的定义的原因所在，社会上对正义黑客的理解是非常模糊的。

本书对前两版中的材料进行了更新，并尝试将最全面、最新的技术、流程和材料汇集起来。因此增加了9章全新的内容，同时对其他章的内容进行了新新。

本书第 部分制定了灰帽黑客必要的伦理和期望基础。

该部分内容包括：理清人们对白帽、黑帽和灰帽黑客的定义和特征的混淆认识 讨论在实施任何类型的正义黑客行动前应该了解的一些棘手的道德问题 讨论漏洞发现报告的难点以及可用于解决这些难题的一些模型 调查黑客攻击以及许多其他类型的恶意行为所涉及的法律问题 概览适当的漏洞发现流程以及当前提供指导方向的模型。

第 部分介绍了现如今其他书籍中没有讲到的更高级的渗透测试方法和工具。

现在的许多书籍讲的都是些相同的、被无数次反复翻新的旧的工具和方法，而本书决定更加深入地讲解真正的灰帽黑客正在使用的高级机制。

该部分我们讨论如下内容： 用来实施渗透测试的自动化渗透测试方法和高级工具 最新的渗透测试工具潜入攻击、社会工程以及内部攻击。

第 部分将深入底层代码，向读者讲解各种操作系统和应用程序的特定部件的工作原理以及如何对它们加以利用。

该部分将讨论如下内容： 介绍一些基本的编程知识，这些是理解剩余内容所需要掌握的概念 如何利用栈操作漏洞以及如何识别和编写缓冲区溢出攻击代码 如何识别高级的Linux和Windows漏洞以及如何对它们加以利用 如何创建不同类型的shellcode来开发自己的概念验证漏洞攻击程序和必要的软件，以测试和识别漏洞 最新的攻击类型，包括基于客户端的、Web服务器、VoIP以及SCADA攻击 第 部分将更深入地研究正义黑客技术的最高级主题，甚至当今的许多安全专家都没有理解这些主题。

该部分将讨论如下内容： 被动和主动分析工具和方法 如何识别源代码和二进制文件中的漏洞 如何进行软件的逆向工程和组件的反汇编 模糊处理和调试技术 如何为二进制和源代码打补丁 第 部分将讲解恶意软件分析。

有时候正义黑客会遇到恶意软件，而且可能需要进行一些基本的分析。

该部分将讨论如下内容： 收集恶意软件样本 分析恶意软件，包括对反模糊处理技术进行讨论。

如果您希望进一步提高和加深对正义黑客技术的了解，那么本书非常适合您。

## &lt;&lt;灰帽黑客&gt;&gt;

## 内容概要

运用最新技术查找和修复安全漏洞，有力地阻止恶意的网络入侵。

《灰帽黑客：正义黑客的道德规范、渗透测试、攻击方法和漏洞分析技术(第3版)》在上一版本的基础上做了全面细致的更新，共新增了9章精彩内容：详细介绍了最新的安全漏洞和补救措施以及法定披露方式，分析黑客们如何定位目标系统、击破保护方案、编写恶意代码以及利用windows和linux系统中的缺陷，还探讨了恶意软件分析、渗透测试、scada、网络电话和web安全等方面的内容。

《灰帽黑客：正义黑客的道德规范、渗透测试、攻击方法和漏洞分析技术(第3版)》主要内容

使用backtrack和metasploit来模拟漏洞攻击

分析潜入、社交工程和内部攻击技术

模拟构建用来发起栈缓冲区溢出攻击的perl、python和ruby脚本

理解并阻止adobe、office和多媒体文件中的恶意内容

检测和阻止客户端、web服务器、voip和scada攻击

介绍针对windows和linux软件的逆向工程、模糊处理及反编译

讨论sql注入、跨站脚本和伪造攻击

使用蜜罐和沙箱技术来捕获恶意软件和rootkit

<<灰帽黑客>>

作者简介

作者：（美国）哈珀（Allen Harper）（美国）哈里斯（Shon Harris）译者：杨明军 韩智文 程文俊 哈珀（Allen Harper），CISSP，曾任美国财政部、国税局计算机安全应急响应中心的安全分析师，现任N2NetSecurity公司总裁Allen经常在Black HatChrechno等会议上发表演讲。

哈里斯（Shon Harris），CISSP，Logical Security总裁、作家、教育工作者、安全顾问。Shon已经发表多篇著作和论文，Information Security Magazine杂志将她评为信息安全领域最杰出的25位女性之一。

## &lt;&lt;灰帽黑客&gt;&gt;

## 书籍目录

## 第一部分“合乎道德的揭秘行为”简介

## 第1章 正义黑客的道德规范

## 1.1 理解敌方策略的意义

## 1.2 认识安全领域的灰色区域

## 1.3 本书与正义黑客类图书的关系

## 1.3.1 漏洞评估

## 1.3.2 渗透测试

## 1.4 关于黑客类图书和课程的争议

## 1.4.1 工具的双重性

## 1.4.2 识别攻击

## 1.4.3 模拟攻击

## 1.5 攻击者最喜欢利用的漏洞

## 第2章 合乎道德的正常揭秘行为

## 2.1 各方看待问题的不同角度

## 2.2 cert目前采取的工作流程

## 2.3 完全揭秘策略—rainforest puppy policy

## 2.4 internet安全组织

## 2.4.1 发现漏洞

## 2.4.2 通知

## 2.4.3 验证

## 2.4.4 解决

## 2.4.5 发布

## 2.5 争议仍将存在

## 2.6 案例分析

## 2.6.1 正当揭秘过程的优缺点

## 2.6.2 供应商更加关注

## 2.7 接下来应该做什么

## 第二部分 渗透测试及工具

## 第3章 社会工程攻击

## 3.1 社会工程攻击原理

## 3.2 实施社会工程攻击

## 3.3 渗透测试中常用到的攻击手段

## 3.3.1 好心人

## 3.3.2 会议室

## 3.3.3 加入公司

## 3.4 准备好进行面对面的攻击

## 3.5 防御社会工程攻击

## 第4章 潜入攻击

## 4.1 潜入攻击如此重要的原因

## 4.2 实施潜入攻击

## 4.2.1 侦察

## 4.2.2 思想准备

## 4.3 进入目标建筑物的常用方法

## 4.3.1 吸烟区入口

## 4.3.2 人工检查点

## &lt;&lt;灰帽黑客&gt;&gt;

## 4.3.3 锁住的门

## 4.3.4 物理方式开锁

## 4.3.5 进入目标之后

## 4.4 防御潜入攻击

## 第5章 内部攻击

## 5.1 模拟内部攻击的重要性

## 5.2 实施内部攻击

## 5.2.1 工具和准备工作

## 5.2.2 了解情况

## 5.2.3 获得本地管理员权限

## 5.2.4 禁用防病毒软件

## 5.2.5 使用cain

## 5.3 防御内部攻击

## 第6章 使用backtrack linux发行

## 6.1 backtrack简介

## 6.2 将backtrack安装到dvd或优盘

## 6.3 直接在虚拟机中使用back track iso映像文件

## 6.3.1 使用virtualbox创建backtrack虚拟机

## 6.3.2 引导backtrack livedvd 系统

## 6.3.3 探索backtrack x窗口环境

## 6.3.4 启动网络服务

## 6.4 永久性更改backtrack

## 6.4.1 将backtrack完整地安装在硬盘或者优盘中

## 6.4.2 新建一个包含永久性更改信息的iso文件

## 6.4.3 使用自定义文件自动保存和恢复更改

## 6.5 研究backtrack引导菜单

## 6.6 更新backtrack

## 第7章 使用metasploit

## 7.1 metasploit简介

## 7.2 获取metasploit

## 7.3 使用metasploit控制台加载攻击工具

## 7.4 使用metasploit攻击客户端漏洞

## 7.5 使用metasploit meterpreter进行渗透测试

## 7.5.1 使用meterpreter进行键盘记录

## 7.5.2 使用meterpreter以不同的已登录用户身份运行代码

## 7.5.3 使用meterpreter的hashdump命令和metasploit的psexec命令以及共享口令登录

## 7.6 metasploit的自动化与脚本化

## 7.7 更进一步探讨metasploit

## 第8章 渗透测试管理

## 8.1 制定渗透测试计划

## 8.1.1 渗透测试的类型

## 8.1.2 渗透测试的范围

## 8.1.3 渗透测试的位置

## 8.1.4 渗透测试小组成员构成

## 8.1.5 方法和标准

## 8.1.6 渗透测试的各个阶段

## 8.1.7 渗透测试计划

## &lt;&lt;灰帽黑客&gt;&gt;

## 8.2 签署渗透测试协议

## 8.2.1 工作声明

## 8.2.2 “保释信”

## 8.3 实施渗透测试

## 8.3.1 测试启动会议

## 8.3.2 渗透测试中的资源访问

## 8.3.3 测试预期值管理

## 8.3.4 测试问题管理

## 8.3.5 欲速则不达

## 8.3.6 外部和内部协同

## 8.4 在渗透测试中进行信息共享

## 8.5 生成渗透测试结果报告

## 8.5.1 报告格式

## 8.5.2 报告摘要

## 第 部分 漏洞攻击

## 第 9 章 编程技能

## 9.1 c编程语言

## 9.1.1 c语言基本结构

## 9.1.2 程序范例

## 9.1.3 使用gcc进行编译

## 9.2 计算机内存

## 9.2.1 随机存取存储器(ram)

## 9.2.2 字节序

## 9.2.3 内存分段

## 9.2.4 内存中的程序

## 9.2.5缓冲区

## 9.2.6内存中的字符串

## 9.2.7 指针

## 9.2.8 内存知识小结

## 9.3 intel处理器

## 9.3.1 寄存器

## 9.4 汇编语言基础

## 9.4.1 机器指令、汇编语言与 c语言

## 9.4.2 at&amp; t与nasm

## 9.4.3 寻址模式

## 9.4.4 汇编文件结构

## 9.4.5 汇编过程

## 9.5 使用gdb进行调试

## 9.5.1 gdb基础

## 9.5.2 使用gdb进行反汇编

## 9.6 python编程技能

## 9.6.1 获取python

## 9.6.2 python中的hello world程序

## 9.6.3 python对象

## 9.6.4 字符串

## 9.6.5 数字

## 9.6.6 列表

## &lt;&lt;灰帽黑客&gt;&gt;

9.6.7 字典

9.6.8 python文件操作

9.6.9 python套接字编程

第10章 基本的linux漏洞攻击

10.1 栈操作

10.1.1 函数调用过程

10.2 缓冲区溢出

10.2.1 meet.c溢出

10.2.2 缓冲区溢出的后果

10.3 本地缓冲区溢出漏洞攻击

10.3.1 漏洞攻击组成部分

10.3.2 在命令行上进行栈溢出漏洞攻击

10.3.3 使用通用漏洞攻击代码进行栈溢出漏洞攻击

10.3.4 对小缓冲区进行漏洞攻击

10.4 漏洞攻击开发过程

10.4.1 控制eip

10.4.2 确定偏移

10.4.3 确定攻击途径

10.4.4 构建漏洞攻击三明治

10.4.5 测试漏洞攻击

第11章 高级linux漏洞攻击

11.1 格式化字符串漏洞攻击

11.1.1 问题描述

11.1.2 从任意内存读取

11.1.3 写入任意内存

11.1.4 利用.dtors获得根特权级

11.2 内存保护机制

11.2.1 编译器改进

11.2.2 内核补丁和脚本

11.2.3 “返回到libc”漏洞攻击

11.2.4 综合比较

第12章 shellcode策略

12.1 用户空间shellcode

12.1.1 系统调用

12.1.2 基本shellcode

12.1.3 端口绑定shellcode

12.1.4 反向shellcode

12.1.5 查找套接字shellcode

12.1.6 命令执行代码

12.1.7 文件传输代码

12.1.8 多级shellcode

12.1.9 系统调用代理shellcode

12.1.10 进程注入shellcode

12.2 其他shellcode考虑因素

12.2.1 shellcode编码

12.2.2 自我破坏shellcode

12.2.3 反汇编shellcode

## &lt;&lt;灰帽黑客&gt;&gt;

- 12.3 内核空间shellcode
  - 12.3.1 内核空间考虑因素
- 第13章 编写linux shellcode
  - 13.1 基本的linux shellcode
    - 13.1.1 系统调用
    - 13.1.2 使用c进行系统调用
    - 13.1.3 使用汇编语言进行系统调用
    - 13.1.4 系统调用exit
    - 13.1.5 系统调用setreuid
    - 13.1.6 利用execve实现创建shell的shellcode
  - 13.2 实现端口绑定shellcode
    - 13.2.1 linux套接字编程
    - 13.2.2 采用汇编语言编程建立一个套接字
    - 13.2.3 测试shellcode
  - 13.3 实现反向连接shellcode
    - 13.3.1 反向连接c语言编程
    - 13.3.2 反向连接汇编程序
  - 13.4 shellcode编码
    - 13.4.1 简单的xor编码
    - 13.4.2 编码后shellcode的结构
    - 13.4.3 jmp/call xor解码器示例
    - 13.4.4 fnstenv xor示例
    - 13.4.5 将代码组合起来
  - 13.5 利用metasploit自动生成shellcode
    - 13.5.1 利用metasploit生成shellcode
    - 13.5.2 利用metasploit对shellcode进行编码
- 第14章 windows漏洞攻击
  - 14.1 windows程序编译与调试
    - 14.1.1 在windows上进行编译
    - 14.1.2 在windows上用ollydbg进行调试
  - 14.2 编写windows漏洞攻击程序
    - 14.2.1 漏洞攻击程序开发过程回顾
    - 14.2.2 prossh服务器
    - 14.2.3 控制eip
    - 14.2.4 确定偏移
    - 14.2.5 确定攻击途径
    - 14.2.6 构建攻击三明治
    - 14.2.7 根据需要调试漏洞攻击程序
  - 14.3 理解seh
    - 14.3.1 seh的实现
  - 14.4 理解windows内存保护(xp sp3、vista、7和server 2008)
    - 14.4.1 基于栈的缓冲区溢出检测(/gs)
    - 14.4.2 safeseh
    - 14.4.3 sehops
    - 14.4.4 堆保护
    - 14.4.5 dep
    - 14.4.6 aslr

## &lt;&lt;灰帽黑客&gt;&gt;

## 14.5 绕过windows内存保护

## 14.5.1 绕过/gs

## 14.5.2 绕过safeseh

## 14.5.3 绕过aslr

## 14.2.4 绕过dep

## 14.5.5 绕过sehops

## 14.5.6 内存保护绕过方法小结

## 第15章 content-type攻击原理与检测

## 15.1 content-type攻击原理

## 15.2 现今可被攻击的文件格式

## 15.3 pdf文件格式简介

## 15.4 恶意pdf漏洞攻击分析

## 15.5 恶意pdf文件检测工具

## 15.5.1 pdfid

## 15.5.2 pdf-parser.py

## 15.6 content-type攻击防御测试工具

## 15.7 content-type攻击防御方法

## 15.7.1 安装所有的安全更新

## 15.7.2 在adobe reader中禁用javascript

## 15.7.3 针对微软office应用程序和adobe reader启用dep

## 第16章 web应用程序安全漏洞

## 16.1 最流行的web应用程序安全漏洞概述

## 16.1.1 注入漏洞

## 16.1.2 跨站脚本漏洞

## 16.1.3 owasp十大隐患中的其他内容

## 16.2 sql注入漏洞攻击

## 16.2.1 sql数据库与语句

## 16.2.2 测试web应用程序，搜寻sql注入漏洞

## 16.3 跨站脚本漏洞攻击

## 16.3.1 “脚本”的含义

## 16.3.2 跨站脚本的含义

## 第17章 oip攻击

## 17.1 voip的含义

## 17.2 voip使用的协议

## 17.2.1 sip

## 17.2.2 megaco h.2

## 17.2.3 h.3

## 17.2.4 tls和dtls

## 17.2.5 srtp

## 17.2.6 zrtp

## 17.3 voip攻击类型

## 17.3.1 枚举

## 17.3.2 sip口令破解

## 17.3.3 窃听与分组捕获

## 17.3.4 拒绝服务

## 17.4 如何防范voip攻击

## 第18章 scada攻击

## &lt;&lt;灰帽黑客&gt;&gt;

- 18.1 scada的含义
- 18.2 scada使用的协议
  - 18.2.1 opc
  - 18.2.2 iccp
  - 18.2.3 modbus
  - 18.2.4 dnp
- 18.3 scada fuzzing测试
  - 18.3.1 使用autodaf é 进行scada fuzzing测试
  - 18.3.2 使用tftp daemon fuzzer进行scada fuzzing测试
- 18.4 stuxnet恶意软件(网络恐怖主义新浪潮)
- 18.5 防范scada攻击
- 第 部分 漏洞分析
- 第19章 被动分析
  - 19.1 道德的逆向工程
  - 19.2 使用逆向工程的原因
    - 19.2.1 逆向工程注意事项
  - 19.3 源代码分析
    - 19.3.1 源代码审计工具
    - 19.3.2 源代码审计工具的实用性
    - 19.3.3 手工源代码审计
    - 19.3.4 自动化源代码分析
  - 19.4 二进制分析
    - 19.4.1 二进制代码的手工审计
    - 19.4.2 自动化的二进制分析工具
- 第20章 使用ida pro进行高级静态分析
  - 20.1 静态分析难点
    - 20.1.1 剥离的二进制文件
    - 20.1.2 静态链接程序和flair
    - 20.1.3 数据结构分析
    - 20.1.4 已编译的c++代码的诡异之处
  - 20.2 扩展ida pro
    - 20.2.1 idc脚本编程
    - 20.2.2 ida pro插件模块及ida pro sdk
    - 20.2.3 构建ida pro插件
    - 20.2.4 ida pro加载器及处理器模块
- 第21章 高级逆向工程技术
  - 21.1 软件攻击的目的
  - 21.2 软件开发过程概述
  - 21.3 检测工具
    - 21.3.1 调试器
    - 21.3.2 代码覆盖分析工具
    - 21.3.3 统计分析工具
    - 21.3.4 流程分析工具
    - 21.3.5 内存使用监视工具
  - 21.4 模糊测试
  - 21.5 定制的模糊测试工具和技术
    - 21.5.1 一个简单的url模糊测试工具

## &lt;&lt;灰帽黑客&gt;&gt;

- 21.5.2 对未知协议进行模糊测试
- 21.5.3 spike
- 21.5.4 spike静态内容原语
- 21.5.5 spike proxy
- 21.5.6 sharefuzz
- 第22章 客户端浏览器的漏洞攻击
- 22.1 客户端软件漏洞的重要性
- 22.1.1 客户端漏洞可以规避防火墙保护
- 22.1.2 客户端应用程序经常在管理权限下运行
- 22.1.3 客户端漏洞易于针对特定人群或机构目标
- 22.2 internet explorer的安全概念
- 22.2.1 activex控件
- 22.2.2 internet explorer安全区域
- 22.3 客户端漏洞攻击的历史与发展趋势
- 22.3.1 客户端漏洞的流行
- 22.3.2 历史上针对客户端攻击的著名漏洞
- 22.4 挖掘基于浏览器的新漏洞
- 22.4.1 mangleme
- 22.4.2 mozilla安全团队的模糊测试工具
- 22.4.3 axenum
- 22.4.4 axfuzz
- 22.4.5 axman
- 22.5 可利用的堆喷射技术
- 22.5.1 internetexploiter
- 22.6 防范客户端漏洞攻击
- 22.6.1 同步更新安全补丁
- 22.6.2 获取最新信息
- 22.6.3 在缩减权限下运行internet应用
- 第23章 攻击windows访问控制模型
- 23.1 攻击访问控制机制的理由
- 23.1.1 多数人不理解访问控制机制
- 23.1.2 访问控制漏洞易于攻击
- 23.1.3 访问控制漏洞数量巨大
- 23.2 windows访问控制的工作机制
- 23.2.1 安全标识符
- 23.2.2 访问令牌
- 23.2.3 安全描述符
- 23.2.4 访问检查
- 23.3 访问控制配置分析工具
- 23.3.1 转储进程令牌
- 23.3.2 转储安全描述符
- 23.4 特殊sid、特殊访问权限和“禁止访问”问题
- 23.4.1 特殊的sid
- 23.4.2 特殊访问权限
- 23.4.3 “禁止访问”的原理
- 23.5 访问控制引起的提权漏洞
- 23.6 各种对象类型的攻击模式

## &lt;&lt;灰帽黑客&gt;&gt;

- 23.6.1 服务攻击
- 23.6.2 windows注册表dacl攻击
- 23.6.3 目录dacl攻击
- 23.6.4 文件dacl攻击
- 23.7 其他对象类型的枚举方法
- 23.7.1 共享内存段
- 23.7.2 命名管道
- 23.7.3 进程
- 23.7.4 其他已命名的内核对象(信号量、互斥锁、事件、设备)
- 第24章 智能模糊测试框架sulley
- 24.1 协议分析
- 24.2 sulley模糊测试框架
- 24.2.1 安装sulley
- 24.2.2 强大的模糊测试工具
- 24.2.3 块结构
- 24.2.4 监视进程中的错误
- 24.2.5 监视网络流量
- 24.2.6 控制vmware
- 24.2.7 综述
- 24.2.8 崩溃事件的事后分析
- 24.2.9 网络使用分析
- 24.2.10 进一步研究
- 第25章 漏洞的可利用性和漏洞攻击程序
- 25.1 漏洞的可利用性
- 25.1.1 通过调试分析可利用性
- 25.1.2 初始分析
- 25.2 理解漏洞攻击问题
- 25.2.1 先决条件和后置条件
- 25.2.2 可重复性
- 25.3 构造漏洞攻击程序有效载荷的有关考虑
- 25.3.1 漏洞攻击程序有效载荷的协议元素
- 25.3.2 缓冲区的方向
- 25.3.3 自毁式shellcode
- 25.4 对问题进行归档
- 25.4.1 背景知识
- 25.4.2 环境
- 25.4.3 研究结果
- 第26章 关闭漏洞：缓解问题
- 26.1 各种缓解方案
- 26.1.1 端口碰撞技术
- 26.1.2 迁移
- 26.2 打补丁
- 26.2.1 对源代码打补丁的注意事项
- 26.2.2 给二进制程序打补丁的注意事项
- 26.2.3 二进制变异
- 26.2.4 第三方打补丁方案
- 第 部分 恶意软件分析

## &lt;&lt;灰帽黑客&gt;&gt;

## 第27章 收集恶意软件和初步分析

## 27.1 恶意软件

## 27.1.1 恶意软件类型

## 27.1.2 恶意软件的防护技术

## 27.2 蜜网技术的最新发展趋势

## 27.2.1 蜜罐

## 27.2.2 蜜网

## 27.2.3 为什么要使用蜜罐

## 27.2.4 蜜罐的局限性

## 27.2.5 低交互性蜜罐

## 27.2.6 高交互性蜜罐

## 27.2.7 蜜网的类型

## 27.2.8 规避vmware检测技术

## 27.3 捕捉恶意软件：设置陷阱

## 27.3.1 vmware宿主机设置

## 27.3.2 vmware客户机设置

## 27.3.3 使用nepenthes进行捕获

## 27.4 恶意软件的初步分析

## 27.4.1 静态分析

## 27.4.2 动态分析

## 27.4.3 norman sandbox技术

## 第28章 破解恶意软件

## 28.1 恶意软件的发展趋势

## 28.1.1 嵌入的组件

## 28.1.2 加密的使用

## 28.1.3 用户空间隐藏技术

## 28.1.4 rootkit技术的应用

## 28.1.5 持久化措施

## 28.2 对恶意软件进行去混淆处理

## 28.2.1 加壳程序基础

## 28.2.2 对二进制文件进行脱壳处理

## 28.3 对恶意软件进行逆向工程

## 28.3.1 恶意软件的设置阶段

## 28.3.2 恶意软件的运行阶段

## 28.3.3 自动化的恶意软件分析

## &lt;&lt;灰帽黑客&gt;&gt;

## 章节摘录

版权页：插图：2.6.2 供应商更加关注 用户期望供应商提供易用且没有错误的软件。

当发现bug后，他们希望供应商立即发布修复措施。

这确实是一把双刃剑。

但是，供应商经常采取的“渗透并修补”的做法招致了安全社区的指责，因为供应商仅仅是发布多个临时性的修复措施来安抚用户，从而避免自己的声誉受损。

安全专家称，这种临时方法不是一种良好的工程实践。

大多数安全缺陷发生在应用程序设计过程的早期。

应用程序的优劣可以通过6个关键的因素区分：身份验证和授权 最好的应用程序应能保证身份验证和授权步骤是完善的，并且不能被绕开。

不信任用户输入 应该将用户当做“敌方代理”，在服务器端验证数据，并通过去掉字符串的标记来防止缓冲区溢出。

端到端会话加密 应该加密整个会话，而不是只加密活动中包含敏感信息的部分。

另外，安全的应用程序的超时时限应该较短，如果用户有一段时间不活动，那么在再次使用应用程序时需要重新进行身份验证。

安全的数据处理 安全的应用程序还将保证系统处于不活动状态时数据是安全的。

例如，口令存储在数据库中时仍然应该是加密的，而且应该实现安全的数据隔离机制。

不恰当地实现加密组件常常为敏感数据的未授权访问提供了很多机会。

消除不当配置、后门和默认设置 许多软件供应商的一种常见但不安全的做法是销售带有后门、实用工具和管理功能的软件，以帮助接收信息的管理员了解和使用该产品，问题在于，这类增强通常会包含严重的安全缺陷。

因此应该总是禁用这些选项，用户可以在需要时再启用它们，并且所有的后门都应该从源代码中恰当地清除。

安全质量保证 设计产品时，无论是在规范制定和开发阶段，还是在测试阶段，安全性都应该是一个核心的关注点。

因此，供应商应该组建安全质量保证团队（security quality assurance team, SQA）来管理所有与安全相关的问题。

2.7 接下来应该处理的事项 我们可以通过完成一些活动来改进当前的安全状况，但是该过程中涉及的每个人都应该更具有前瞻性、接受过良好的培训并具有更大的动力。

如果确实想让自己的环境更加安全，那么就应该遵循下面列出的一些惯例：积极行动 确保环境安全不只是开发人员的责任，同时也是用户的责任，用户应该积极查阅有关安全功能的文档，并向供应商索取测试结果。

许多安全问题就是因为用户配置不当造成的。

培训应用程序开发人员 经过良好培训的开发人员可以创建更安全的产品。

供应商应该在安全领域认真地培训他们的雇员。

#### 媒体关注与评论

如果您是一名渗透测试人员或研究人员，而且希望快速地提高和拓宽自己的IT安全技能，那么本书将是您的良师益友。

——Corelan Team创始人Peter Van Eeckhoutte ( corelanc0d3r )

## <<灰帽黑客>>

### 编辑推荐

《灰帽黑客:正义黑客的道德规范、渗透测试、攻击方法和漏洞分析技术(第3版)》是由清华大学出版社出版。

## <<灰帽黑客>>

### 名人推荐

“一本面向渗透测试者和研究员的优秀参考书，将帮助他们在广阔的信息安全领域提高和扩展技巧。”  
——Corelan Team创始人Peter Van EeCkhoutte(corelancOd3r)

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>