Broadview
www.broadview.com.cn

安全技术大系

看雪软件安全
http://www.pediy.com

0day安全：软件漏洞分析技术

failwest 编著

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

0day安全：软件漏洞分析技术

failwest 编著

電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

本书分为 4 篇 17 章，系统全面地介绍了 Windows 平台缓冲区溢出漏洞的分析、检测与防护。第一篇为常用工具和基础知识的介绍；第二篇从攻击者的视角出发，揭秘了攻击者利用漏洞的常用伎俩，了解这些知识对进行计算机应急响应和提高软件产品安全性至关重要；第三篇在第二篇的基础上，从安全专家的角度介绍了漏洞分析和计算机应急响应方面的知识；第四篇则站在软件工程师的角度讲述如何在开发、测试等软件生命周期的各个环节中加入安全因素，以增强软件产品的安全性。

图书在版编目（CIP）数据

0 day 安全：软件漏洞分析技术 / 王清编著.—北京：电子工业出版社，2008.4
ISBN 978-7-121-06077-9

I. 0… II.王… III.软件可靠性—基本知识 IV.TP311.5

中国版本图书馆 CIP 数据核字（2008）第 023638 号

策划编辑：毕　宁
责任编辑：韩　明
印　　刷：北京市铁成印刷厂
装　　订：北京中新伟业印刷有限公司
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　邮编 100036
开　　本：787×980　1/16　　印张：23.5　　字数：410 千字
印　　次：2008 年 4 月第 1 次印刷
印　　数：5000 册　　定价：49.00 元　（含光盘 1 张）

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：（010）88258888。

关于“zero day attack”

0 day 是网络安全技术中的一个术语，特指被攻击者掌握却未被软件厂商修复的系统漏洞。

0 day 漏洞是攻击者入侵系统的终极武器，资深的黑客手里总会掌握几个功能强大的 0 day 漏洞。

0 day 漏洞是木马、病毒、间谍软件入侵系统的最有效途径。

由于没有官方发布的安全补丁，攻击者可以利用 0 day 对目标主机为所欲为，甚至在 Internet 上散布蠕虫。因此，0 day 漏洞的技术资料通常非常敏感，往往被视为商业机密。

对于软件厂商和用户来说，0 day 攻击是危害最大的一类攻击。

针对 0 day 漏洞的缓冲区溢出攻击是对技术性要求最高的攻击方式。

世界安全技术峰会 Black Hat 上每年最热门的议题之一就是“zero day attack/defense”。微软等世界著名的软件公司为了在其产品中防范“zero day attack”，投入了大量的人力、物力。

全世界有无数的信息安全科研机构在不遗余力地研究与 0 day 安全相关的课题。

全世界也有无数技术精湛的攻击者在不遗余力地挖掘软件中的 0 day 漏洞。

自　序

不请长缨，系取天骄种，剑吼西风

——《六州歌头》北宋，贺铸

虽然事隔多年，我仍然清晰记得自己被“冲击波”愚弄的场景——2003 年夏的那个晚上，自己像往常一样打开实验室的计算机，一边嘲笑着旁边同学因为不装防火墙而被提示系统将在一分钟内关机，一边非常讽刺地在自己的计算机上发现了同样的提示对话框。正是这个闻名世界的“框框”坚定了我投身网络安全研究的信念，而漏洞分析与利用正是这个领域的灵魂所在。

漏洞分析与利用的过程是充满艺术感的。想象一下，剥掉 Windows 中那些经过层层封装的神秘的对话框“外衣”，面对着浩如烟海的二进制机器码，跋涉于内存中不知所云的海量数据，在没有任何技术文档可以参考的情况下，进行反汇编并调试，把握函数调用和参数传递的细节，猜测程序的设计思路，设置巧妙的断点并精确定位到几行有逻辑缺陷的代码，分析研究怎么去触发这个逻辑漏洞，最后编写出天才的渗透代码，从而得到系统的控制权……这些分析过程的每一个环节无不散发着充满智慧的艺术美感！这种技术不同于其他计算机技术，它的进入门槛很高，需要拥有丰富的计算机底层知识、精湛的软件调试技术、非凡的逻辑分析能力，还要加上一点点创造性的思维和可遇而不可求的运气。

在无数个钻研这些技术的夜里，我深深地感觉到国内的漏洞分析资料和文献是多么匮乏。为了真正搞清楚蠕虫病毒是怎样利用 Windows 漏洞精确淹没 EIP 寄存器并获得进程控制权，我仍然记得自己不得不游走于各种论坛收集高手们零散手稿时的情形。那时的我多么希望能有一本教材式的书籍，让我读了之后比较全面、系统地了解这个领域。

我想，在同样漆黑的夜里，肯定还有无数朋友和我从前一样，满腔热情地想学习这门技术而又困惑于无从下手。正是这种“请缨无处，剑吼西风”的感觉，激励着我把自己钻研的心血凝结成一本教程，希望这样一本教程可以帮助喜欢网络安全的朋友们在学习时绕开我曾走过的弯路。

failwest

2008 年 3 月 1 日

推荐序

很久以来，没有人愿意公开地去研究软件及系统的漏洞。应该说 failwest 是少数几个真正从软件开发者的角度著书阐述漏洞分析与检测技术的专业软件工程师。作者非常恰当地把着眼点放在一个软件开发者的角度去做漏洞检测，使得《0 day 安全：软件漏洞分析技术》对大多数读者来说更加实用。

《0 day 安全：软件漏洞分析技术》为我们系统介绍了漏洞分析的原理和技术细节，并深入浅出地引用了不少在安全界非常经典的漏洞实例。然而，更重要的是 failwest 并没有流水账式的罗列知识与技术，而是花了大量的篇幅介绍了漏洞检测的步骤及其背后的思维方式。这些完全不同的思维方式，加上分析员必备的技能以及必需的工具，为读者展现了一套非常完整的软件漏洞分析方法。

许　明

前　言

关于安全技术人才

国内外对网络安全技术人才的需求量很大，精通缓冲区溢出攻击的安全专家可以在大型软件公司轻易地获得高薪的安全咨询职位。

信息安全技术是一个对技术性要求极高的领域，除了扎实的计算机理论基础外，更重要的是优秀的动手实践能力。在我看来，不懂二进制数据就无从谈起安全技术。

国内近年来对网络安全的重视程度正在逐渐增加，许多高校相继成立了“信息安全学院”或者设立“网络安全专业”。科班出身的学生往往具有扎实的理论基础，他们通晓密码学知识、知道 PKI 体系架构，但要谈到如何真刀实枪地分析病毒样本、如何拿掉 PE 上复杂的保护壳、如何在二进制文件中定位漏洞、如何对软件实施有效的攻击测试……能够做到的人并不多。

虽然每年有大量的网络安全技术人才从高校涌入人力市场，真正能够满足用人单位需求的却寥寥无几。捧着书本去做应急响应和风险评估是滥竽充数的作法，社会需要的是能够为客户切实解决安全风险的技术精英，而不是满腹教条的阔论者。

我所认识的很多资深安全专家都并非科班出身，他们有的学医、有的学文、有的根本没有学历和文凭，但他们却技术精湛，充满自信。

这个行业属于有兴趣、够执著的人，属于为了梦想能够不懈努力的意志坚定者。

关于“Impossible”与“I’m possible”

从拼写上看，“Impossible”与“I’m possible”仅仅相差一个用于缩写的撇号（apostrophe）。学完本书之后，您会发现将“不可能（Impossible）”变为“可能（I’m possible）”的“关键（key point）”往往就是那么简单的几个字节，本书将要讨论的就是在什么位置画上这一撇！

从语法上看，“Impossible”是一个单词，属于数据的范畴；“I’m possible”是一个句子，含有动词（算符），可以看成是代码的范畴。学完本书之后，您会明白现代攻击技术的精髓就是混淆数据和代码的界限，让系统错误地把数据当作代码去执行。

从意义上看，To be the apostrophe which changed “Impossible” into “I’m possible” 代表着人类挑战自我的精神，代表着对理想执著的追求，代表着对事业全情的投入，代表着敢于直面惨淡人

生的豪情……而这一切正好是黑客精神的完美诠释——还记得在电影《Sword Fish（剑鱼行动）》中，Stan 在那台酷毙的计算机前坚定地说："Nothing is impossible"，然后开始在使用 Vernam 加密算法和 512 位密钥加密的网络上，挑战蠕虫的经典镜头吗？

于是我在以前所发表过的所有文章和代码中都加入了这个句子，甚至用它作为自己的签名档。

尽管我的英语老师和不少外国朋友提醒我，说这个句子带有强烈的"Chinglish"味道，甚至会引起 Native Speaker 的误解，然而我最终还是决定把它写进书里。

虽然我不是莎士比亚那样的文豪，可以创造语言，发明修辞，用文字撞击人们的心灵，但这句"Chinglish"的确能把我所要表达的含义精确地传递给中国人，这已足够。

关于本书

通常情况下，利用缓冲区溢出漏洞需要深入了解计算机系统，精通汇编语言乃至二进制的机器代码，这足以使大多数技术爱好者望而却步。

随着时间的推移，缓冲区溢出攻击在漏洞的挖掘、分析、调试、利用等环节上已经形成了一套完整的体系。伴随着调试技术和逆向工程的发展，Windows 平台下涌现出的众多功能强大的 debug 工具和反汇编分析软件逐渐让二进制世界和操作系统变得不再神秘，这有力地推动了 Windows 平台下缓冲区溢出的研究。除此以外，近年来甚至出现了基于架构（Frame Work）的漏洞利用程序开发平台，让这项技术的进入门槛大大降低，使得原本高不可攀的黑客技术变得不再遥不可及。

遗憾的是，与国外飞速发展的高级黑客技术相比，目前国内还没有系统介绍 Windows 平台下缓冲区溢出漏洞利用技术的专业书籍，而且相关的中文文献资料也非常匮乏。

本书将系统全面地介绍 Windows 平台软件缓冲区溢出漏洞的发现、检测、分析和利用等方面的知识。

为了保证这些技术能够被读者轻松理解并掌握，本书在叙述中尽量避免枯燥乏味的大段理论阐述和代码粘贴。概念只有在实践中运用后才能真正被掌握，这是我多年来求学生涯的深刻体会。书中所有概念和方法都会在紧随其后的调试实验中被再次解释，实验和案例是本书的精髓所在。从为了阐述概念而精心自制的漏洞程序调试实验到现实中已经造成很大影响的著名漏洞分析，每一个调试实验都有着不同的技术侧重点，每一个漏洞利用都有自己的独到之处。

我将带领您一步一步地完成调试的每一步，并在这个过程中逐步解释漏洞分析思路。不管您是网络安全从业人员、黑客技术发烧友、网络安全专业的研究生或本科生，如果您能够完成这些分析实验，相信您的软件调试技术、对操作系统底层的理解等计算机能力一定会得到一次质的飞跃，并能够对安全技术有一个比较深入的认识。

内容导读

本书分为 4 篇，共 17 章。

第 1 篇　基础知识

第 1 章　漏洞概述

简介漏洞研究中的一些基础概念和原理

第 2 章　二进制文件概述

不管是漏洞挖掘，漏洞分析还是漏洞利用，我们所面对的都是二进制、机器码、内存地址。第 2 章将简单介绍 Windows 平台下可执行文件的结构和内存方面的一些基础知识。PE 文件和虚拟内存的细节枯燥乏味，长篇累牍地介绍很容易让人失去学习的兴趣和激情。但在进行静态反汇编和动态调试的过程中，如果没有 PE 和虚拟内存方面的基础知识，您甚至无法把反汇编的内容和正在执行的指令对应起来。根据漏洞分析的特点，这章给出了调试漏洞所必须的二进制基础知识。

第 3 章　必备工具

第 3 章介绍了一批漏洞分析中经常使用的软件工具。包括调试工具、反汇编工具、二进制编辑工具等。您会在后面的调试实验中反复见到这些工具的身影。在这章的最后一节，我设计了一个非常简单的破解小实验，用于实践工具的应用，消除您对二进制的恐惧感，希望能够给您带来一些乐趣。

第 2 篇　漏洞利用

第 4 章　栈溢出利用

基于栈的溢出是最基础的漏洞利用方法。第 4 章首先用大量的示意图，深入浅出地讲述了操作系统中函数调用、系统栈操作等概念和原理；随后通过三个调试实验逐步讲解如何通过栈溢出，一步一步地劫持进程并植入可执行的机器代码。即使您没有任何汇编语言基础，从未进行过二进制级别的调试，在本章详细的实验指导下也能轻松完成实验，体会到 exploit 的乐趣。

第 5 章　开发 shellcode 的艺术

第 5 章紧接第 4 章的讨论，比较系统地介绍了溢出发生后，如何布置缓冲区、如何定位 shellcode、如何编写和调试 shellcode 等实际的问题。第 5 章的最后两小节还给出了一些编写 shellcode 的高级技术，供有一定汇编基础的朋友参考。

第 6 章　堆溢出利用

在很长一段时间内，Windows 下的堆溢出被认为是不可利用的，然而事实并非如此。第 6 章将用精辟的论述点破堆溢出利用的原理，让您轻松领会堆溢出的精髓。此外，这章的一系列调试实验将加深您对概念和原理的理解。用通俗易懂的方式论述复杂的技术是本书始终坚持的原则。

第 7 章　Windows 异常处理机制深入浅出

对异常处理的利用是 Windows 平台下缓冲区溢出漏洞利用的一大特点。第 7 章除了介绍如何在溢出发生时利用 S.E.H 外，还对 Windows 异常处理机制做了较深入的剖析，供有一定基础的读者参考。

第 8 章　高级内存攻击技术

集中介绍了一些曾发表于 Black Hat 上的著名论文中所提出的高级利用技术。对于安全专家，了解这些技巧和手法不至于在分析漏洞时错把可以利用的漏洞误判为低风险类型；对于黑客技术爱好者，这些知识很可能成为激发技术灵感的火花。

第 9 章　揭秘 Windows 安全机制

微软在 Windows XP SP2 和 Windows 2003 之后，向操作系统中加入了许多安全机制。本章将集中讨论这些安全机制对漏洞利用的影响。

第 10 章　用 MetaSploit 开发 Exploit

MetaSploit 是软件工程中的 Frame Work（架构）在安全技术中的完美实现，它把模块化、继承性、封装等面向对象的特点在漏洞利用程序的开发中发挥得淋漓尽致。使用这个架构开发 Exploit 要比直接使用 C 语言写出的 Exploit 简单得多。第 10 章将集中介绍如何使用这个架构进行 Exploit 开发，这也将是第一次在中文书籍中集中介绍 MetaSploit 通用漏洞测试平台。

第 11 章　其他漏洞利用技术

格式化串漏洞在 Windows 平台上非常罕见，所以我把这种漏洞利用单独放在本章介绍。除此以外，由于脚本注入漏洞与缓冲区溢出漏洞的攻防在技术上差异较大，故也被安排在这章。

鉴于基于 Web 的漏洞利用种目繁杂，且自成体系，本书目前只做了简单的介绍。如有机会，我将单独著书述之。

第 3 篇　漏洞分析

第 12 章　漏洞分析技术概述

第 12 章纵览了漏洞分析与调试的思路，并介绍了一些辅助漏洞调试分析的高级逆向工具。

第 13 章　MS06-040 分析：系统入侵与蠕虫

通过对真实案例的分析，彻底揭秘攻击者入侵操作系统的全过程。在您获得操作系统控制权限的那一刻，相信伴随着强烈的成就感，您也将切身体会 0day 的真正危害和安全补丁的重要性。

第 14 章　MS06-055 分析：揭秘“网马”

通过网页“挂马”是近年来攻击者惯用的手法。本章通过分析微软 IE 浏览器中真实的缓冲区溢出漏洞，告诉您为什么不能随便点击来历不明的 URL 链接。

第 15 章　MS07-060 分析：Word 文档中的阴谋

稍懂计算机知识的人都不会随便点击可执行文件，但是谁会想到打开 Word 文档也会导致 shellcode 的执行呢？如果 Office 中存在漏洞，那么打开 Word 文档就也有可能导致 shellcode 被执行。

第 4 篇　漏洞挖掘与软件安全性测试

第 16 章　漏洞挖掘技术浅谈

不论从工程上讲还是从学术上讲，漏洞挖掘都是一个相当前沿的领域。本章将介绍一些目前比较流行的漏洞挖掘方法，并着重介绍 Fuzz 测试的方法。相信本章的内容对于 QA 工程师和软件测试人员也会有用。

第 17 章　安全的软件生命周期

要做到尽可能地避免软件中的安全漏洞，仅仅靠安全测试和漏洞挖掘是远远不够的，那需要在软件生命周期的各个环节中加入安全因素。

本书源代码及相关文档

本书中调试实验所涉及的所有源代码和 PE 文件都被收录在附带光盘中。

这些代码都经过了仔细调试，如在使用中发现问题，请查看实验指导中对实验环境的要求。个别攻击实验的代码可能会被部分杀毒软件鉴定为存在风险的文件，请您调试前详细阅读实验说明。

此外，“看雪论坛”相关版面可以找到更多本书中所涉及的资源：http://zeroday.pediy.com

对读者的要求

虽然溢出技术经常涉及汇编语言，但本书并不要求读者一定具备汇编语言的开发能力。所用到的指令和寄存器在相关的章节都有额外介绍，只要您有 C 语言基础就能消化本书的绝大部分内容。

我并不推荐在阅读本书之前先去系统的学习汇编知识和逆向知识，枯燥的寻址方式和指令介绍很容易让人失去学习的兴趣。本书将带您迅速跨过漏洞分析与利用技术的进入门槛。即使您并不懂汇编与二进制也能完成书中的调试实验，并获得一定的乐趣。当然，在您达到一定水平想进一步提高时，补习逆向知识和汇编语言将是绝对必要的。

本书适合的读者群体包括：

- **安全技术工作者** 本书比较全面、系统地收录了 Windows 平台下缓冲区溢出攻击所涉及的各种方法，将会是一本不错的技术字典。
- **信息安全理论研究者** 本书中纰漏的许多漏洞利用、检测方法在学术上具有一定的前沿性，在一定程度上反映了目前国内外安全技术所关注的焦点问题。
- **QA 工程师、软件测试人员** 本书第 4 篇中集中介绍了产品安全性测试方面的知识，这些方法可以指导 QA 人员审计软件中的安全漏洞，增强软件的安全性，提高软件质量。
- **软件开发人员** 知道漏洞利用原理将有利于编写出安全的代码。
- **高校信息安全专业的学生** 本书将在一定程度上弥补高校教育与信息安全公司人才需求脱节的现象。用一套过硬的调试技术和逆向技术来武装自己可以让您在未来的求职道路上利于不败之地。精通 exploit 的人才可以轻松征服任何一家杀毒软件公司或安全资讯公司的求职门槛，获得高薪工作。
- **本科二年级以上计算机系学生** 通过调试实验，你们将更加深入地了解计算机体系架构和操作系统。这些知识一样将成为您未来求职时过硬的敲门砖。
- **所有黑客技术爱好者** 如果您厌倦了网络嗅探、端口扫描之类的扫盲读物，您将在本书中学到实施有效攻击所必备的知识和技巧。

反馈与提问

读者在阅读本书时如遇到任何问题，可以到看雪论坛相关版面提出或发送 E-mail 给我。

致谢

感谢电子工业出版社对本书的大力支持，尤其是毕宁与韩明编辑为本书出版所做的大量工作。

感谢看雪对本书的大力推荐和支持以及看雪论坛为本书提供的交流平台。

感谢 Dafydd Stuttard 和 Matthew Conover 在我编写 shellcode 技术和堆溢出技术的相关章节时提供的热情帮助。你们不但拥有精湛的技术，难能可贵的共享精神也是我学习的榜样。

感谢金山毒霸反病毒引擎组给我提供的实习机会，尤其感谢 Bani Cai、大灰、涂老师和 Zmworm，是你们带我跨过了逆向技术的门槛。

感谢赛门铁克产品安全部的同事，尤其是我的经理 Cassio Goldschmid。宽松的技术氛围和一流的技术培训为我的成长提供了强有力的支持，一起去参加 Black Hat 对我来说胜过任何节日。

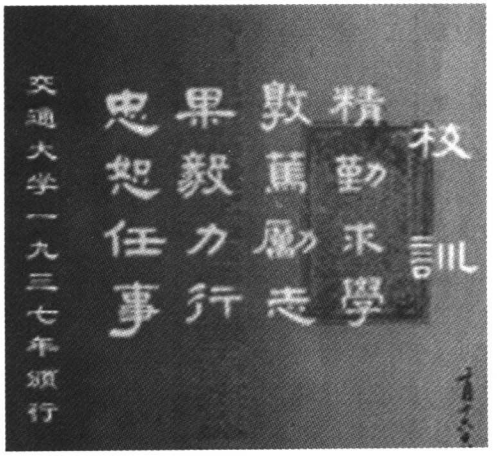
感谢“下一代互联网和网络安全国家重点实验室”，尤其感谢我的导师管晓宏教授，是您为我提供了开放的研究环境和丰富的研究资源。

感谢 114 实验室的兄弟姐妹，和你们在一起的日子我非常开心。

感谢我的爸爸、妈妈，谢谢你们的理解与支持。

感谢司徒雪岚，我会永远记得著书期间那段让我欢喜让我忧的日子。

最后感谢我的母校西安交通大学，是那里踏实求是的校风与校训激励着我不断进步。



目　录

第 1 篇　基础知识

第 2 篇　漏洞利用

第 3 篇 漏洞分析