

# 软件漏洞分析技术

吴世忠 郭涛 董国伟 张普含 著



科学出版社

# 软件漏洞分析技术

吴世忠 郭 涛 著  
董国伟 张普含

科学出版社

北 京

## 内 容 简 介

本书通过对概念的规范、方法的归纳、实例的分析和技术的对比,从白盒到黑盒、从静态到动态、从理论到实践、从传统到新兴,全面深入地阐述了软件漏洞分析技术的各方面内容,是国内第一部系统性介绍漏洞分析技术的著作。

本书共分5部分17章。其中,第1部分介绍软件漏洞的基本概念、影响和发展历程,并提出漏洞分析技术的总体框架;第2部分~第4部分从源代码漏洞分析、二进制漏洞分析、软件架构与运行系统漏洞分析三个维度对软件漏洞分析技术进行了深入阐述;第5部分为前沿技术及未来展望,对移动智能终端、云计算、物联网和工控系统等新兴领域中的漏洞分析技术及挑战进行了总结,探讨了漏洞分析未来发展方向。

本书面向从事信息安全、软件安全和漏洞分析学习、研究、实践和管理的各类专业人士,为他们提供了理论的指南、实践的指导和趋势的指引,是一本必备的宝典,也可作为信息安全相关专业师生教材,以及IT专业培训教材。

### 图书在版编目(CIP)数据

软件漏洞分析技术 / 吴世忠等著. —北京: 科学出版社, 2014.9

ISBN 978-7-03-041890-6

I . 软… II . ①吴… III . 软件可靠性 IV . TP311.53

中国版本图书馆 CIP 数据核字 (2014) 第 211384 号

责任编辑: 杨 凯 / 责任制作: 魏 谨

责任印制: 赵德静 / 封面设计: 于启宝

北京东方科龙图文有限公司 制作

<http://www.okbook.com.cn>

科学出版社出版

北京东黄城根北街16号

邮政编码: 100717

<http://www.Longmenbooks.com>

骏杰印刷有限公司 印刷

科学出版社发行 各地新华书店经销

\*

2014年11月第一版 开本: 787×1092 1/16

2014年11月第一次印刷 印张: 33 1/2

字数: 770 000

定 价: 88.00

(如有印装质量问题, 我社负责调换)



# 前言

随着社会信息化进程的不断推进和互联网的日益普及，计算机及信息技术在国民经济和社会生活的各个领域中都得到了广泛的应用。软件，作为计算机系统功能实现的重要载体，更是提供了丰富多彩的信息化体验。然而，随着社交网络、微博、移动互联网、云计算、物联网等新技术、新应用的不断出现，由软件引发的信息安全事件也层出不穷，“震网病毒”、“火焰病毒”事件的发酵引发了人们对国家关键信息基础设施保护的关注；“维基泄密”等事件引起了人们对网络上个人隐私的担忧；而斯诺登披露的“棱镜门”事件更是把信息安全问题推到了风口浪尖。分析发现，所有这些信息安全事件都存在一个共同点——信息系统或软件自身存在可被利用的漏洞，软件安全漏洞是绝大多数信息安全事件的根源。通过深入探究安全漏洞的本质可以发现在信息技术研发、应用的整个生命周期中，漏洞广泛存在，而且几乎不可避免。首先，冯·诺依曼计算机体系自身的缺陷以及基于TCP/IP协议栈的互联网体系的不安全性等问题，导致了安全漏洞的不可避免性；其次，随着计算机软件系统规模的快速增长以及新技术和新应用的推陈出新，软件系统影响范围的扩展和复杂度的提高，增大了产生漏洞的概率；最后，软件开发生命周期的各个环节都有人为参与，实践经验的缺乏和防范意识的疏忽都有可能導致安全漏洞。

“千里之堤，溃于蚁穴”，漏洞对网络信息安全的重要性和关键性不言自明，以漏洞挖掘、漏洞检测、漏洞应用、漏洞消除和漏洞管控等为主要内容的漏洞分析工作在信息安全保障工作中的重要性和基础性同样不言而喻。

中国信息安全测评中心长期致力于信息安全漏洞分析和风险评估工作，在国家相关部委和社会各界的支持下，结合国家信息安全的战略需求和现实需要，积极开展漏洞分析专项工作，取得了可喜的成效。已有的漏洞分析技术主要针对软件的源代码形态和二进制形态。在现代软件开发环境下，通常将源代码编译或解析成二进制代码，因此源代码中的安全缺陷更可能会直接导致软件漏洞的产生，从而引发重大信息安全事件；针对源代码的漏洞分析较为直观，但是无法发现编译、链接过程中引入的漏洞，同时出于商业利益保护等原因，大部分软件的源代码难以获得，因此，面向二进制程序的漏洞分析技术应用范围也较为广泛。我们的工作实践表明，漏洞分析除了涵盖软件代码分析范畴以外，还应对需求规约、架构设计等开发文档，以及实际运行系统开展安全性分析。架构安全性分析可以发现软件设计方面存在的缺陷，从而在较早的阶段发现安全问题；而软件发布后，在配置、部署、运行、维护和升级等过程中会有意或者无意地引入一些漏洞，因此还要对运行系统进行漏洞分析，以保证其安

全性。本书通过对概念的规范、方法的归纳、实例的分析和技术的对比，从白盒到黑盒、从静态到动态、从理论到实践、从传统到新兴，试图对软件漏洞分析技术的各个方面作一阶段性总结。同时还针对新技术、新应用、新产品的出现分析了漏洞分析所面临的机遇和挑战，并探讨该领域未来的发展趋势。

本书共分为5部分17章。其中，第1部分介绍了软件漏洞的基本概念、影响和发展历程，并提出漏洞分析技术的总体框架；第2部分至第4部分从源代码漏洞分析、二进制漏洞分析、软件架构与运行系统漏洞分析三个维度对软件漏洞分析技术进行了深入阐述；第5部分为前沿技术及未来展望，对移动智能终端、云计算、物联网和工控系统等新兴领域中的漏洞分析及挑战进行总结，探讨漏洞分析未来发展方向。信息化发展前景无限，网络空间精彩纷呈，信息安全至关重要，漏洞分析任重道远，我们深感这项工作充满“道高一尺，魔高一丈”的对抗和挑战，我们的技术探索和工作努力不能懈怠，不可止息。本书只是漏洞分析的一个方面，相关内容还没有经过更广的实践和更长时间的检验，我们诚挚欢迎得到各界专家学者的建议和批评。我们希望能够通过本书为从事信息安全和漏洞分析学习、研究、实践和管理的人员提供帮助，为漏洞分析技术的突破和创新提供支撑，并吸引更多的有志之士参与到漏洞分析工作中来，以增强我国的信息安全保障能力。

在本书编写过程中，中国信息安全测评中心的王欣、邵帅、辛伟、朱龙华、张翀斌、贾依真、时志伟、郝永乐、王眉林、柳本金、邓辉，北京邮电大学的梁洪亮和崔宝江老师、解放军信息工程大学的魏强老师、中国人民大学的梁彬和石文昌老师、中国科技大学的程绍银老师、北京神州绿盟科技有限公司安全研究部的左磊高级研究员和忽朝俭博士、湖北大学的张葵老师等给予了大力支持。此外，中国科学院软件研究所的冯登国研究员和张健研究员、中国科学院信息工程研究所的荆继武教授和邹维研究员、北京大学的陈钟教授和王千祥教授、北京邮电大学的宫云战教授，中国信息安全测评中心的章磊、王嘉捷、侯元伟、康凯、刘德、连一汉、覃日鸿、吴健雄、苏权、郑亮、杨诗雨、张丹、崔晓雷，以及北京科技大学的甄平、哈尔滨工业大学的景慧昀、北方工业大学的曲珑玉和北京邮电大学的朱京晶等也对本书提出了宝贵的意见。在此，对他们一并表示由衷的感谢！

本书得到了中国信息安全测评中心“漏洞分析与风险评估”专项工程、国家自然科学基金项目(61272493、61100047)和国家高技术研究发展技术(863计划)项目(2012AA012903)的支持。

吴世忠

2014年7月于北京上地

# 目 录

## 第1部分 漏洞分析基础

### 第1章 软件中的漏洞

1.1 漏洞概述 .....	3
1.1.1 漏洞定义 .....	3
1.1.2 漏洞特点 .....	7
1.2 漏洞分类与分级 .....	9
1.2.1 漏洞分类 .....	9
1.2.2 漏洞分级 .....	12
1.3 漏洞的影响 .....	15
1.3.1 漏洞无处不在 .....	15
1.3.2 漏洞分析保障信息安全 .....	18
参考文献 .....	22

### 第2章 漏洞分析发展历程

2.1 软件漏洞分析 .....	25
2.1.1 广义漏洞分析 .....	26
2.1.2 狭义漏洞分析 .....	26
2.2 原始萌芽阶段 .....	28
2.2.1 通信安全 .....	28
2.2.2 分析萌芽 .....	29
2.2.3 信息加密 .....	30
2.3 初步发展阶段 .....	30
2.3.1 计算机安全 .....	31
2.3.2 单一化漏洞挖掘 .....	31
2.3.3 操作系统防护 .....	32

2.4 高速发展阶段 .....	33
2.4.1 互联网安全 .....	33
2.4.2 多样化漏洞分析 .....	34
2.4.3 信息系统防护 .....	36
2.5 综合治理阶段 .....	38
2.5.1 网际安全 .....	38
2.5.2 系统化漏洞管控 .....	39
2.5.3 防御体系建设 .....	44
参考文献 .....	48

### 第3章 漏洞分析技术概述

3.1 漏洞分析技术体系 .....	56
3.2 软件架构安全分析 .....	58
3.2.1 形式化架构分析技术 .....	59
3.2.2 工程化架构分析技术 .....	60
3.2.3 分析技术对比 .....	60
3.3 源代码漏洞分析 .....	61
3.3.1 基于中间表示的分析技术 .....	63
3.3.2 基于逻辑推理的分析技术 .....	64
3.3.3 分析技术对比 .....	64
3.4 二进制漏洞分析 .....	66
3.4.1 静态漏洞分析技术 .....	67
3.4.2 动态漏洞分析技术 .....	68
3.4.3 动静结合的漏洞分析技术 .....	68
3.4.4 分析技术对比 .....	69
3.5 运行系统漏洞分析 .....	70
3.5.1 信息收集 .....	71
3.5.2 漏洞检测 .....	71
3.5.3 漏洞确认 .....	72
3.5.4 分析技术对比 .....	72
参考文献 .....	73

## 第2部分 源代码漏洞分析

### 第4章 数据流分析

4.1 基本原理 .....	79
----------------	----



4.1.1 基本概念 .....	79
4.1.2 检测程序漏洞 .....	80
4.1.3 辅助支持技术 .....	83
4.2 方法实现 .....	85
4.2.1 使用数据流分析检测程序漏洞 .....	85
4.2.2 数据流分析作为辅助技术 .....	98
4.3 实例分析 .....	101
4.3.1 使用数据流分析检测程序漏洞 .....	101
4.3.2 数据流分析作为辅助技术 .....	106
4.4 典型工具 .....	107
4.4.1 Fortify SCA .....	107
4.4.2 Coverity Prevent .....	109
4.4.3 FindBugs .....	112
参考文献 .....	114

## 第 5 章 污点分析

5.1 基本原理 .....	116
5.1.1 基本概念 .....	116
5.1.2 使用污点分析技术挖掘程序漏洞 .....	117
5.2 方法实现 .....	119
5.2.1 基于数据流的污点分析 .....	119
5.2.2 基于依赖关系的污点分析 .....	125
5.3 实例分析 .....	126
5.4 典型工具 .....	128
5.4.1 Pixy .....	128
5.4.2 TAJ .....	130
参考文献 .....	131

## 第 6 章 符号执行

6.1 基本原理 .....	134
6.1.1 基本概念 .....	134
6.1.2 检测程序漏洞 .....	136
6.1.3 构造测试用例 .....	138
6.1.4 与其他漏洞分析技术结合 .....	140
6.2 方法实现 .....	141



6.2.1 使用符号执行检测程序漏洞 .....	141
6.2.2 使用符号执行构造测试用例 .....	148
6.3 实例分析 .....	149
6.3.1 检测程序漏洞 .....	149
6.3.2 构造测试用例 .....	151
6.4 典型工具 .....	153
6.4.1 Clang .....	153
6.4.2 KLEE .....	156
参考文献 .....	158

## 第 7 章 模型检测

7.1 基本原理 .....	161
7.1.1 基本概念 .....	161
7.1.2 技术框架 .....	162
7.1.3 方法特点 .....	163
7.2 方法实现 .....	164
7.2.1 程序建模 .....	165
7.2.2 安全缺陷属性描述 .....	166
7.2.3 程序漏洞检查 .....	168
7.3 实例分析 .....	170
7.3.1 线性时序逻辑检查 .....	170
7.3.2 分布式软件的漏洞检测 .....	172
7.4 典型工具 .....	175
7.4.1 SLAM .....	176
7.4.2 MOPS .....	178
参考文献 .....	182

## 第 8 章 定理证明

8.1 基本原理 .....	185
8.1.1 基本概念 .....	185
8.1.2 技术框架 .....	186
8.1.3 方法特点 .....	187
8.2 方法实现 .....	187
8.2.1 程序转换 .....	188
8.2.2 属性描述 .....	192

8.2.3 定理证明 .....	194
8.3 实例分析 .....	198
8.3.1 整数溢出漏洞分析 .....	198
8.3.2 TLS协议逻辑漏洞分析 .....	200
8.3.3 输入验证类漏洞分析 .....	203
8.4 典型工具 .....	205
8.4.1 Saturn .....	205
8.4.2 ESC/Java .....	207
参考文献 .....	208

## 第3部分 二进制漏洞分析

### 第9章 模糊测试

9.1 基本原理 .....	215
9.1.1 基本概念 .....	215
9.1.2 基本过程 .....	216
9.2 方法实现 .....	217
9.2.1 输入数据的关联分析 .....	217
9.2.2 测试用例集的构建方法 .....	223
9.2.3 测试异常分析 .....	227
9.2.4 模糊测试框架 .....	231
9.3 实例分析 .....	232
9.3.1 文件模糊测试 .....	233
9.3.2 网络协议模糊测试 .....	236
9.3.3 ActiveX控件模糊测试 .....	239
9.4 典型工具 .....	242
9.4.1 Peach .....	242
9.4.2 Sulley .....	244
参考文献 .....	246

### 第10章 动态污点分析

10.1 基本原理 .....	247
10.1.1 基本概念 .....	247
10.1.2 动态污点分析原理 .....	248
10.1.3 方法特点 .....	250

10.2 方法实现 .....	251
10.2.1 污点数据标记 .....	252
10.2.2 污点动态跟踪 .....	254
10.2.3 污点误用检查 .....	260
10.3 实例分析 .....	262
10.4 典型工具 .....	264
10.4.1 TaintCheck .....	265
10.4.2 Argos .....	265
10.4.3 TaintDroid .....	267
参考文献 .....	268
<b>第 11 章 基于模式的漏洞分析</b>	
11.1 基本原理 .....	270
11.1.1 基本概念 .....	270
11.1.2 二进制文件结构 .....	271
11.2 方法实现 .....	277
11.2.1 反汇编分析 .....	277
11.2.2 逆向中间表示 .....	280
11.2.3 漏洞模式建模 .....	289
11.2.4 漏洞模式检测 .....	293
11.3 实例分析 .....	295
11.3.1 缓冲区溢出类漏洞实例（不安全函数调用） .....	295
11.3.2 缓冲区溢出类漏洞实例（循环写内存） .....	298
11.3.3 整数溢出类漏洞实例 .....	299
11.3.4 内存地址对象破坏性调用类漏洞实例 .....	300
11.4 典型工具 .....	302
11.4.1 BinNavi .....	302
11.4.2 BAP .....	303
参考文献 .....	304
<b>第 12 章 基于二进制代码比对的漏洞分析</b>	
12.1 基本原理 .....	307
12.1.1 研究现状 .....	307
12.1.2 基本原理 .....	308
12.2 方法实现 .....	310

12.2.1 基于文本的二进制代码比对 .....	310
12.2.2 基于图同构的二进制比对 .....	311
12.2.3 基于结构化的二进制比对 .....	313
12.2.4 软件补丁的二进制比对技术 .....	317
12.3 实例分析 .....	318
12.3.1 漏洞信息搜集 .....	318
12.3.2 搭建调试环境 .....	318
12.3.3 补丁比对 .....	319
12.3.4 静态分析 .....	321
12.3.5 动态调试 .....	322
12.4 典型工具 .....	324
12.4.1 Bindiff .....	324
12.4.2 Eye Binary Diffing Suite .....	326
参考文献 .....	328

## 第 13 章 智能灰盒测试

13.1 基本原理 .....	330
13.1.1 基本概念 .....	330
13.1.2 智能灰盒测试的过程 .....	331
13.2 方法实现 .....	332
13.2.1 动态符号执行 .....	333
13.2.2 路径控制与定向遍历 .....	339
13.2.3 路径约束求解 .....	350
13.2.4 漏洞触发数据生成 .....	359
13.3 实例分析 .....	364
13.3.1 关键路径提取 .....	364
13.3.2 中间符号设置 .....	365
13.3.3 污点分析 .....	366
13.3.4 执行Trace获取 .....	368
13.3.5 路径控制与定向遍历 .....	369
13.3.6 约束求解 .....	370
13.3.7 结论 .....	371
13.4 典型工具 .....	371
13.4.1 Dart .....	371
13.4.2 Smart fuzzing .....	373

13.4.3 BitBlaze .....	375
参考文献 .....	376

## 第4部分 架构安全和运行系统漏洞分析

### 第14章 软件架构安全分析

14.1 基本原理 .....	383
14.1.1 软件架构定义 .....	383
14.1.2 架构安全概述 .....	384
14.1.3 架构安全分析过程 .....	386
14.2 方法实现 .....	386
14.2.1 形式化分析技术 .....	386
14.2.2 工程化分析技术 .....	391
14.3 实例分析 .....	400
14.3.1 形式化分析实例——UMLsec .....	400
14.3.2 工程化分析实例——基于Web应用程序的威胁建模 .....	403
14.4 典型工具 .....	406
14.4.1 威胁建模工具 .....	406
14.4.2 软件架构分析工具 .....	408
参考文献 .....	409

### 第15章 运行系统漏洞分析

15.1 基本原理 .....	411
15.1.1 基本概念 .....	411
15.1.2 运行系统漏洞分析 .....	412
15.2 方法实现 .....	414
15.2.1 信息收集 .....	414
15.2.2 配置管理漏洞检测 .....	417
15.2.3 通信协议漏洞检测 .....	418
15.2.4 授权认证漏洞检测 .....	420
15.2.5 数据验证漏洞检测 .....	423
15.2.6 数据安全性漏洞检测 .....	425
15.3 实例分析 .....	426
15.3.1 信息收集 .....	426
15.3.2 漏洞检测过程 .....	426

15.4 典型工具 .....	429
15.4.1 Nmap .....	429
15.4.2 Nessus .....	430
15.4.3 微软基线安全分析器 .....	431
15.4.4 WVS .....	432
参考文献 .....	434

## 第5部分 前沿技术及未来展望

### 第16章 漏洞分析领域新挑战

16.1 移动智能终端漏洞分析 .....	439
16.1.1 领域背景 .....	439
16.1.2 漏洞分析技术 .....	440
16.1.3 应对措施 .....	443
16.2 云计算平台漏洞分析 .....	444
16.2.1 领域背景 .....	444
16.2.2 漏洞分析技术 .....	446
16.2.3 应对措施 .....	449
16.3 物联网漏洞分析 .....	449
16.3.1 领域背景 .....	449
16.3.2 漏洞分析技术 .....	451
16.3.3 应对措施 .....	455
16.4 工控系统漏洞分析 .....	456
16.4.1 领域背景 .....	456
16.4.2 漏洞分析技术 .....	457
16.4.3 应对措施 .....	461
16.5 其他新兴领域的漏洞分析 .....	462
16.5.1 智能家居的漏洞分析 .....	462
16.5.2 智能交通的漏洞分析 .....	464
16.5.3 可穿戴设备的漏洞分析 .....	466
参考文献 .....	467

### 第17章 漏洞分析技术展望

17.1 理论突破 .....	476
17.1.1 软件模型构建 .....	476

17.1.2 漏洞模式提取 .....	477
17.1.3 技术极限求解 .....	478
17.2 技术发展 .....	479
17.2.1 精确度判定 .....	479
17.2.2 分析性能的提高 .....	480
17.2.3 智能化提升 .....	481
17.3 工程实现 .....	482
17.3.1 大规模软件的漏洞分析 .....	482
17.3.2 新型平台上的漏洞分析 .....	485
17.3.3 通用平台级漏洞分析软件开发 .....	488
参考文献 .....	489

## 附录A 国内外重要漏洞库介绍

A.1 国家级漏洞库 .....	492
A.2 行业级漏洞库 .....	495
A.3 民间级漏洞库 .....	496
参考文献 .....	502

## 附录B 国内外重要安全标准介绍

B.1 总体情况 .....	503
B.2 基础类标准 .....	504
B.2.1 国内标准 .....	504
B.2.2 国际标准 .....	505
B.3 防范类标准 .....	509
B.3.1 国内标准 .....	509
B.3.2 国际标准 .....	512
B.4 管理类标准 .....	514
B.4.1 国内标准 .....	514
B.4.2 国际标准 .....	515
B.5 应用类标准 .....	517
B.5.1 国际标准 .....	517



## 漏洞分析基础

随着全球信息化的迅猛发展，计算机软件已成为世界经济、科技、军事和社会发展的重要引擎。在当前日益严峻的信息安全大环境下，信息窃取、资源被控、系统崩溃等各类信息安全事件层出不穷，给国民经济、国家安全、社会稳定等带来严重威胁。究其根源，所有这些信息安全事件都存在一个共同点，那就是信息系统或软件自身存在可被利用的漏洞。因此，漏洞分析日益成为信息安全领域理论研究和实践工作的焦点，越来越引起世界各国的关注与重视。

软件产业的蓬勃发展是信息化应用能够如此丰富的主要原因，但同时也带来了软件漏洞数量的激增，近年来，由于软件漏洞被恶意利用而引发的重大信息安全事件越来越多。因此，了解软件中漏洞的基本概念和特性，熟悉漏洞分析的经验和方法，对减少漏洞的危害、降低信息系统运营成本将会起到良好作用，进而尽量降低重大信息安全事件发生的概率。

本部分从定义、分类、分级、管理及所具备的突出特点等方面介绍软件漏洞的基本概念，并说明软件漏洞的巨大危害；在此基础上，分四个阶段说明学术界、企业界、各国政府机构等一直以来在漏洞挖掘、检测、消除和管控等方面进行的研究和开展的工作；最后分别以软件架构、源代码、二进制代码和运行系统为对象，从整体上分析现有漏洞分析技术的框架，并从基本原理、优缺点和应用范围等方面对它们进行对比。

