反编译技术与软件逆向分析

FANBIANYI JISHU
YU RUANJIAN NIXIANG FENXI

赵荣彩 庞建民 张靖博 编著

国防工业出版社
National Defense Industry Press

FANBIANYI JISHU
YU RUANJIAN NIXIANG FENXI

反编译技术与软件逆向分析

FAN BIAN YI JI SHU YU RUAN JIAN NI XIANG FEN XI

赵荣彩　庞建民　张靖博　编著

国防工业出版社

·北京·

内 容 简 介

本书共分10章。第1章到第3章简要介绍了软件逆向分析技术的相关基础知识；第4章和第5章从反汇编和中间表示两个方面为反编译奠定基础；第6章到第9章针对反编译的若干关键技术展开详细介绍；第10章则为反编译测试相关的一些可用资源。全书以IA-64可执行代码为例进行讲解，但相关技术可以向其他平台推广。

本书可作为计算机软件专业本科高年级学生、硕士研究生的相关课程教科书或教学参考书，也可供从事软件逆向分析工作的工程技术人员参考。

图书在版编目(CIP)数据

反编译技术与软件逆向分析／赵荣彩，庞建民，张靖博编著.
北京：国防工业出版社，2009.11
ISBN 978-7-118-06546-6

Ⅰ.反... Ⅱ.①赵... ②庞... ③张... Ⅲ.①编译程序-程序设计 ②软件工程-分析程序 Ⅳ.TP314 TP311.5

中国版本图书馆CIP数据核字(2009)第162475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)
北京奥鑫印刷厂印刷
新华书店经售
*
开本710×960 1/16 印张14¼ 字数252千字
2009年11月第1版第1次印刷 印数1—3000册 定价36.00元

(本书如有印装错误，我社负责调换)

国防书店：(010)68428422 发行邮购：(010)68414474
发行传真：(010)68411535 发行业务：(010)68472764

编委会名单

主　编　赵荣彩

副主编　庞建民　张靖博

编　者　赵荣彩　庞建民　张靖博

付　文　汪　淼　吴伟峰

前　言

随着计算机科学和相关技术的不断发展，尤其是各种编程语言的不断丰富与壮大，相关人员对贴近于硬件层的低级形式编码越来越陌生。但是，事实表明计算机软件领域从来没有过，也不可能真正脱离对繁琐的低级代码进行分析的需求，而软件逆向分析技术在近年来重新成为计算机科学领域的研究热点。在众多逆向分析技术中，反编译是对目标程序分析最为彻底，但也是最为困难的技术领域。从名称上可以看出，反编译技术是编译技术的逆过程，即将低级目标可执行代码翻译为语义等价的高级语言表示形式。

本书希望能够为从事软件逆向分析的科研人员和工作者提供有效的帮助。与其他相关的逆向分析书籍不同的是，编者没有局限于对二进制代码的反汇编分析，或者局限于对不同逆向分析辅助工具的使用指导，而是希望能够在反汇编层面分析的基础上对目标低级程序进行进一步挖掘，从而获取更多的有效信息。毕竟现有的各种逆向分析工具的功能不一，不能完全满足业界复杂多变的需求。“授人以鱼，不如授之以渔”，本书希望能够帮助读者深入了解并掌握一个完整反编译工具的各个部分，从而编写真正满足自己需求的逆向分析工具。

全书分为三大部分，共10章。第一部分，包括第1章至第3章。简要介绍了软件逆向分析技术的相关基础知识，为读者的进一步阅读奠定良好的基础。包括软件逆向分析的背景知识、不同体系结构指令系统的相关背景，以及针对可执行文件格式的介绍。在指令系统一章中介绍了两种完全不同的体系结构，即CISC体系结构和EPIC体系结构，并且着重针对Intel公司的64位安腾处理器的IA-64体系结构指令系统进行分析。在可执行文件格式一章，则着重解析了在Linux操作系统中流行的ELF可执行文件格式。第二部分，包括第4章、第5章。从反汇编和中间表示两个方面讲述程序分析技术。该部分内容起到了承前启后的作用，不仅帮助读者掌握如何分析二进制机器指令编码，并且介绍了逆向分析中常用的中间表示形式，从而帮助读者在进

一步学习过程中积累必要的基础知识。第三部分为本书的精华，包括第6章至第10章。基本数据类型分析一章是在一些明确定义类型的基本类型的基础上，分析其他没有类型的变量，在格理论的框架之下寻找语句之间的相互关系，从而完成数据类型推导。高级控制流恢复一章介绍如何从线性的二进制可执行文件中挖掘隐藏的高级控制结构，比如条件分支结构和循环结构。过程恢复技术一章则分别针对库函数和用户自定义函数展开，着重分析了在过程分析中涉及到的过程边界分析、参数/返回值分析等内容。部分编译优化效果的消除一章的内容并非试图降低程序的执行效率，而是在逆向分析的过程中，某些特定的编译优化效果在很大程度上将会阻碍逆向分析的正常进行，因此需要将优化的代码还原，该章围绕部分相关技术进行讨论。最后一章介绍了常用的程序调试工具和逆向分析工具，并简要介绍了基本使用方法。

本书由解放军信息工程大学信息工程学院计算机科学与技术系赵荣彩教授带领相关课题组成员，在前期科研实践的基础之上完成的。杨克峤、张雪萌、孙维新、崔平非、朱晓珺、苏铭、齐宁、丁松阳等同志在攻读学位的同时也参与了部分章节的编写或准备工作，在此表示诚挚的感谢。在编写过程中，各级领导都对我们的工作表示了关心和帮助，并在各个方面大力支持编写工作的开展；不少教授、研究生为本书的最终完稿提出了很多很好的建议和帮助，在此表示衷心的感谢。由于编者水平有限，加之时间仓促，书中难免存在缺点和不足之处，甚至可能存在错误。殷切希望各位读者能够批评指正，给我们提出中肯、宝贵的意见。

本书可作为计算机软件专业本科高年级学生、硕士研究生的教材或教学参考书。

编 者

目 录

第1章 绪 论

在某种开发环境下使用高级编程语言进行软件开发，已经成为当今广为人知的软件开发方法。但是，目标软件的分析工作往往无法基于高级源代码展开，而需要针对已经经过编译的，甚至是优化的低级代码进行。这种低级代码可以是以二进制形式表示的机器码指令序列，也可以是被某种虚拟机接受的中间语言。需要指出的是，二进制代码是大多数软件在计算机系统中的保存载体，也是计算机软件得以运行的保证。因此，针对目标二进制代码的软件逆向分析工作具有一定的需求和特有的优势。然而，虽然对于软件逆向分析技术的研究已经有将近50年的历史，但至今仍然没有形成一套良好的分析方法，存在的辅助工具也是针对一些比较简单或特殊的领域。因此，对于软件逆向分析方面的研究还有很长的路要走。

1.1 软件逆向分析

软件逆向分析是一系列对运行于机器之上的低级代码进行等价的提升和抽象，最终得到更加容易被人所理解的表现形式的过程。简言之，软件逆向分析的目的就在于从目标软件中找出设计思想。如图1-1所示，可以将这种分析过程看作一个黑盒子，其输入是可以被处理器理解的机器码表示形式，输出则可以表现为图表、文档，甚至源程序代码等多种形式。

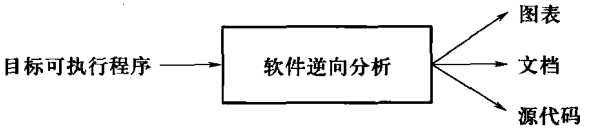

图1-1 软件逆向分析工程

软件逆向分析具有很高的实用价值，包括反编译系统的开发、程序的理解和维护、代码变换系统的正确性验证、恶意代码检测等。软件技术是当前最为复杂，但也是最吸引人的技术之一，软件逆向分析就是关于如何打开一个软件的“黑盒子”，并且探个究竟的过程。当然，我们并不需要什么类似螺丝刀之类的工具帮忙，而只需要一台计算机和人类的智慧。软件逆向分析需要综合使用多

种技能，并且需要对计算机系统以及软件开发流程有比较深刻的理解。但是正如其他大多数意义非凡的领域一样，真正需要的前提条件是强烈的好奇心和求知欲。软件逆向分析集成了许多方面的知识，如代码破解、迷惑解析、编程以及逻辑分析等。出于不同的目的，人们使用不同的分析过程，这些过程都将在本书中进行进一步的探讨。

软件逆向分析对厂商开发出具有竞争力的产品意义非凡，这样说并不过分，实际上，这也是众所周知的软件逆向分析的应用场合。有趣的是，逆向分析并不像我们想象的那样在业界广泛流行。造成这种情况的原因有很多，最主要的是，软件的复杂性使得在很多情况下为了竞争的目的而逆向分析一个软件系统的过程相当复杂，从而造成了沉重的财务负担。那么，什么是软件世界中最为常见的对应用程序的逆向分析呢？主要有两种：与安全相关的逆向分析和与软件开发相关的逆向分析。

1.1.1 与安全相关的逆向分析

对于一些读者来说，将安全与逆向分析联系在一起可能难以理解。而实际上，逆向分析与计算机安全的很多方面都有关联。例如，可以在密码研究中使用逆向分析——研究人员对加密产品进行逆向分析，从而对它提供的安全级别进行评估；在恶意软件的开发与防护中，逆向分析也占据了重要的位置；软件破解也在很大程度上需要逆向分析的帮助，破解者使用这种分析方法来分析，甚至击溃不同的版权保护机制。所有这些应用都将在下面的章节中讨论。

1. 恶意软件

互联网的出现和流行完全改变了计算机业界，尤其是与安全相关方面的应用。数以百万的用户通过互联网相互连接在一起，他们每天都在使用电子邮件等方式进行联系，但是，像病毒、蠕虫这样的恶意软件的传播速度也大大加快。仅仅是十几年前，病毒程序往往需要被复制到一张磁碟之上，而且这张磁碟必须被装载在另外一台计算机上，这样才能够完成病毒传播的过程。由于传播途径的限制，这种感染的过程是相当缓慢的，其防护也相对简单很多。目前，互联网几乎连接了所有的计算机系统，而各种恶意程序已经可以自动地在成千上万的计算机之间悄无声息地传播。正如前文提到的，软件逆向分析在这条恶意软件链的两端都有用武之地。恶意软件开发者使用逆向分析方法定位操作系统和其他软件的弱点。这些弱点能够用来穿透系统的防护层，从而允许通过互联网完成感染操作。除此之外，某些网络黑客还使用逆向分析技术定位软件的弱点，从而允许恶意程序获取关键信息的访问权，甚至完全控制整个系统。在这条链的另一端，反病毒软件开发人员对其获取的每一个恶意程序进行解剖和分析。他

们使用逆向分析技术追踪程序执行的每个步骤，对它们可能造成的危害和可能的感染机率进行评估，研究如何从被感染的系统中清除恶意程序，以及判定是否可以完全避免被感染。

2. 逆向分析加密算法

加密系统往往与隐私有关：一个人传递给另一个人的信息可能并不想让其他人知道。可以粗略地将加密算法分为两组：有限加密算法和基于密钥的算法。有限加密算法好比一些孩子们玩的游戏：写给一个朋友一封信，信中的每个字母都经过向上或向下的若干次移动。有限加密算法的秘密在于算法本身，一旦算法被揭露，也就毫无秘密可言。由于逆向分析可以分析出加密或解密算法，因此有限加密算法只能提供非常脆弱的安全性。由于其算法也是保密的，因此逆向分析可以被看作是对算法的破解过程。

另一方面，基于密钥的算法的秘密是密钥，即一些类似于数字的值，它们可以由某些算法来对信息进行加密和解密。在基于密钥的算法中，用户使用密钥对信息进行加密，并保证密钥的隐蔽性。这种算法通常是公开的，而仅需要保护密钥即可。由于算法是已知的，因此逆向分析变得毫无意义。为了对一条经过基于密钥算法加密的信息进行解密，可能需要以下 3 种途径：①获取密钥；②尝试所有可能的组合；③寻找算法中的缺陷，从而解析出密钥或最初的信息。

尽管如此，对于基于密钥加密方法的逆向分析在某些方面却意义非凡。即便加密算法广为人知，特定的实现细节也会对程序提供的所有安全级别造成意想不到的影响。无论加密算法如何精巧，很小的实现错误也有可能使该算法提供的安全级别失效。而确认一个安全产品是否真正地实现一个加密算法只有两种途径：要么分析它的源代码（假定是可行的），要么进行逆向分析。

3. 数字版权管理

现代计算机系统已经将大多数类型的具有版权的材料转变为数字信息，包括音乐、影视，甚至书籍。这些信息以前只能够通过具体的媒介获取，而现在可以通过数字化信息得到。这种趋势为用户提供了巨大的好处，也为版权拥有者和内容提供商带来了一些问题。对于用户来说，这意味着资料质量的提高，并且易于获取和管理。对于提供商来说，这使得他们能够以很低的费用提供高质量的内容，但更为重要的是，这种方式使得对内容流向的控制无法完成。

数字化信息以难以想象的速度在流动，并且易于复制。这种流动性意味着一旦带有版权的资料到达用户手中，用户能够很容易地对其进行移动和复制，因此盗版也变得相当容易。通常软件公司通过在软件产品中嵌入复制保护技术防止被盗版，即通过在软件产品中嵌入代码片段来防止或限制用户对程序进行复制。

近年来，随着数字多媒体变得越来越流行，媒体内容提供商已经开发或购买了一些技术来控制音乐、视频等内容的发布。这些技术统称为数字版权管理(DRM)技术。从概念上来说，DRM 技术与传统软件复制保护技术非常相似。不同之处在于，软件系统中，作者能够主动提供保护，并且能够决定是否允许它自身的发布。数字多媒体则是被动的，它往往通过其他程序进行播放或阅读，从而更加难以控制或限制其使用。

这个问题与逆向分析也是有很大关系的，这是因为破解者通常使用逆向分析技术来尝试突破 DRM 技术。通过使用逆向分析技术，破解者能够学习这些技术内部的秘密，从而发现如何尽量小地改动程序便可以关闭保护机制。

4. 二进制代码审核

开放源代码软件的一个强项在于它往往天生就更加可靠和安全。无论它提供的安全性究竟如何，总比运行那些由上千个软件工程师检查和证明过的软件“感觉”要安全很多。而且毋庸置疑的是，开源软件也提供了一些真切的质量保证。对于开源软件来说，开放对程序源代码的访问权限意味着那些弱点和安全漏洞能够很快被发现，通常是在恶意程序利用它们之前。对于那些无法获取源代码的私有软件来说，逆向分析成为一项可行的，也是非常有限的安全漏洞查找的备选方案。当然，逆向分析无法得到像开源软件那样的易于操作和可读的信息，但是利用高超的逆向分析技术能够对其二进制代码进行正确性检查，并且评估其造成的不同安全威胁。

1.1.2 针对软件开发的逆向分析

逆向分析对于软件开发者来说也是非常有用的。例如，软件开发者可以利用逆向分析技术发现如何与缺乏文档或只有部分文档的软件之间完成交互。另外，逆向分析技术也能够用来评定类似代码库，甚至评定像操作系统这样的第三方代码的质量。最后，有时也可能使用逆向分析技术从竞争者的产品中提取出有用的信息，从而达到提升自身技术实力的目的。如何在软件开发中应用逆向分析技术将在下面章节中进行讨论。

1. 与私有软件之间的交互

软件工程师几乎每天都可以从逆向分析技术中获得与私有软件交互方面的利益。平时，在使用一个私有库或操作系统 API 时，其文档往往并不充分。无论库的生产者使用了多少种方案来确保所有可能的情况都已经被包含在文档中，用户还是能够发现很多棘手的难以回答的问题。大多数开发者要么坚持尝试不同的方法来完成预定的工作，要么联系生产者来寻求答案。另一方面，那些拥有逆向分析技巧的人却时常发现他们很容易处理这种情况。使用

逆向分析技术使得他们有可能在很短的时间内解决相关问题，并且只需要付出很少的劳动。

2. 开发竞争产品

在大多数产业中，对应用产品的逆向分析是迄今为止最为流行的应用。而软件正在成为大多数产品中最为复杂的产品之一，因此为了创建一个富有竞争力的产品而对其他软件产品完成整体逆向分析则没有任何意义。相比而言，从头设计并开发一个产品，或者仅仅是获得第三方软件组件的支持，而不是闭门造车，这样可能会更加容易一些。在软件工业中，即使竞争对手拥有一个没有获得专利的技术，对他们的整个产品进行逆向分析也不会有什么帮助。独立开发属于自己的软件总是要容易很多。但凡事总有例外，即对于那些难以完成开发的高度复杂或特殊的设计和算法，大多数模块仍然应当独立开发，但对于高度复杂或不寻常的模块，则应该使用逆向分析技术进行分析。

3. 评估软件质量和鲁棒性

与审核程序的二进制码从而对其安全性和漏洞进行评估类似，我们也可能会尝试抽取部分程序的二进制码来对代码实现的质量进行评估。该需求非常简单：开源软件就像一本开放的书籍，允许它的用户在造成威胁之前对它的质量进行评估。软件开发商往往不会发布他们软件的源代码，他们希望用户能够“仅仅信任他们”。

这种对于核心软件产品（例如操作系统）的源代码访问权的要求在一些大型公司的应用中犹为明显。几年之前，微软就已经宣布了大型用户购买了超过1000个获得访问Windows源代码的名额，从而对其进行评估。对于那些缺乏足够的购买力以访问产品源代码的用户来说，要么相信宣传中所说的“产品是安全的”，要么对其进行逆向分析。另外，逆向分析永远也无法像查阅源代码那样揭示产品的代码质量和总体可靠性，但是依然能够提供丰富的信息。此处并不需要太多特殊的技术。一旦习惯于使用逆向分析方法，将能够非常快速地浏览二进制代码，能够使用这种能力尝试评估其质量。

1.1.3 本书的主要内容

软件逆向分析所涵盖的领域是相当广泛的，不仅包括一些常用逆向分析工具的使用，而且包含了许多非常具有创造性的工作人员的智慧。但是，由于逆向分析的目标不同，因此对于目标领域的熟悉程度在很大程度上影响了分析的效果。例如，在加密程序的分析中，如果熟悉相关加密算法则能够快速帮助分析人员定位目标程序所使用的算法类型，从而决定如何完成下一步的分析工作。但是计算机在各个领域中都有着非常广泛而深入的应用，因此不可能列举出所有

领域的经验知识。基于这种情况，本书不可能也无法总结并归纳出所有逆向分析经验。但是，逆向分析的过程中却有着很多相似的概念和方法。例如，通常分析的过程包含与系统相关的分析和独立于系统的分析。与系统相关的分析，是指在分析中需要大量的系统相关知识辅助，对于不同系统之间的分析差异很大。独立于系统的分析，是指当把目标程序抽象到某一个级别表示之后，便能够抛开具体系统的影响，而专注于更加具有创造性的研究过程之中。本书对于这两种类型的分析都将会有所涉及，但主要定位在如何完成从与系统相关到与系统无关的提升。这个过程类似于一种称为反编译器的逆向分析工具的工作过程。通常来讲，源代码是与系统无关的，至少不会受限于硬件处理器的特殊规格信息。反编译器的目的就是希望能够将可执行目标代码自动翻译为等价的高级语言代码。本书的内容将紧扣反编译器的一般实现过程，之所以这样处理，是因为相关知识不仅能够帮助用户加深对程序运行和逆向分析概念的理解，其中涉及的很多知识也能够直接应用于软件逆向分析之上。

1.2 软件逆向分析的历史

严格来讲，软件逆向分析的历史应当与计算机程序的历史相当，早期的计算机程序是使用二进制指令直接编写的，一旦读取二进制指令，理解目标程序的实现概念的过程，便可以称为软件逆向分析过程。随着第一个 Fortran 语言编译器的出现，程序员再也不用动手操作二进制指令或与之等价的汇编指令了，程序的开发转向以高级语言为主的开发。与此同时，人们对于低级代码的阅读和理解能力也在不断下降。但是这种下降之中却存在着一些隐患。例如，对于没有源代码的程序分析变得异常困难，对于编译器的安全性和正确性要求非常之高。因此，在第一个编译器出现之后不久，便出现了以恢复程序高级源代码为目的的反编译器。

1.3 软件逆向分析的各个阶段

前面已经提到，软件逆向分析可以分为与系统相关的分析和独立于系统的分析，但这种划分并不十分精确，而且在某些案例分析的过程中，这两种分析之间是相互交叉的。可以说，对于一个完整软件系统的分析过程是上述两种分析不断循环重复的过程。图 1-2 显示了常见软件逆向分析的各个阶段，包括文件装载、指令解码、语义映射、相关图构造、过程分析、类型分析、结果输出 7 个阶段。一般来说，软件逆向分析的过程根据其不同的目的可以选取若干阶段来完