“十一五”国家重点图书　　计算机科学与技术学科前沿丛书
计算机科学与技术学科研究生系列教材（中文版）

软件保护技术

王建民　王朝坤　余志伟　编著

清华大学出版社

“十一五”国家重点图书　　计算机科学与技术

计算机科学与技术学科研究生系列教材（中文版）

软件保护技术

王建民　王朝坤　余志伟　编著

清华大学出版社
北京

内 容 简 介

本书较为系统地介绍现有的软件保护技术，共分 7 章，主要包括软件保护概述、软件保护的技术基础、软件水印技术、软件混淆技术、软件防篡改技术、软件保护技术的综合使用以及软件保护技术的总结与展望，旨在为解决软件版权问题提供新的思路。本书在理论讲解的基础上，也提供了若干具体操作实例，通俗易懂，便于读者理解和实践。

本书适合作为高等学校高年级本科生以及研究生相关课程的教材和自学教材，也可作为计算机软件安全研究和开发的指导参考书。

图书在版编目(CIP)数据

软件保护技术 / 王建民，王朝坤，余志伟编著. —北京：清华大学出版社，2013
(计算机科学与技术学科前沿丛书)
计算机科学与技术学科研究生系列教材(中文版)
ISBN 978-7-302-31721-0

Ⅰ. ①软… Ⅱ. ①王… ②王… ③余… Ⅲ. ①软件—安全技术—研究生—教材 Ⅳ. ①TP311.56

中国版本图书馆 CIP 数据核字(2013)第 048728 号

责任编辑：焦　虹　战晓雷
封面设计：傅瑞学
责任校对：时翠兰
责任印制：杨　艳

出版发行：清华大学出版社
网　　址：http://www.tup.com.cn，http://www.wqbook.com
地　　址：北京清华大学学研大厦 A 座　　邮　　编：100084
社 总 机：010-62770175　　邮　　购：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质 量 反 馈：010-62772015，zhiliang@tup.tsinghua.edu.cn

印 刷 者：北京富博印刷有限公司
装 订 者：北京市密云县京文制本装订厂
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　　张**：8.5　　**字　　数**：204 千字
版　　次：2013 年 7 月第 1 版　　**印　　次**：2013 年 7 月第 1 次印刷
印　　数：1～2000
定　　价：25.00 元

产品编号：037767-01

前言

随着计算机网络技术的飞速发展，软件产品的版权保护以及内容安全性成为一个亟待解决的问题，越来越多地受到人们的普遍关注。软件保护技术的发展以及相关法律法规的健全，可以在很大程度上保障软件的分发和运行安全。其中，设计和提出切实可行的保护技术显得尤为重要。以软件水印、软件混淆和软件防篡改技术为主的软件保护技术正是为解决软件知识产权保护问题而发展起来的一个新兴学科。

本书内容安排如下：第1章为引言部分，介绍软件保护的背景、意义以及研究现状；第2章介绍软件保护的技术基础，包括代码操作技术、程序分析技术、Java字节码的结构以及拆分和编码知识；第3章至第5章为本书的重点，分别介绍软件水印技术、软件混淆技术和软件防篡改技术；第6章讨论上述若干保护技术的综合使用；第7章为总结和展望。

本书在写作过程中得到了清华大学软件学院硕士研究生张长江、付军宁、李嘉、王潇等同学的帮助，在此表示感谢。

尽管我们对全书内容进行了全面的修订和校正，但由于水平有限，书中可能会存在一些问题和不足，恳请广大读者提出宝贵意见和建议。

编　者

2013年3月

软件水印技术的相关符号表

P	应用程序
W	水印
F	指纹
ε	空水印
P'	添加水印后/混淆后/篡改保护后的应用程序
K	密钥
$\mathbf{P}$	应用程序的集合
$\mathbf{W}$	水印的集合
$\mathbf{K}$	密钥的集合
B	基本块
Em	嵌入器
Ex	提取器
Dec	检测器
Reg	识别器

目录

第1章 引　言

1.1 软件保护的背景及意义

随着计算机技术和因特网技术的飞速发展，数字产品，特别是软件产品的版权侵犯、非法复制和恶意篡改等行为日趋泛滥，这不仅给软件产品的生产商、分销商和服务商带来极大的负面影响，还影响了政府的税收收入，甚至引发新的计算机安全问题等。因此，软件产品的版权保护以及内容安全性成为一个亟待解决的问题。

如图 1-1 所示，根据商用软件联盟(Business Software Alliance，BSA)和国际数据公司(International Data Corporation，IDC)公布的 2008 年全球盗版研究报告[1]，2008 年全球 PC 软件盗版率为 41%，其造成的直接经济损失达 53 亿美元。报告还显示，在被调查的 110 个国家中，盗版率最高的国家是亚美尼亚、孟加拉、格鲁吉亚和津巴布韦，其盗版率均超过 90%，盗版率最低的是美国、日本、新西兰和卢森堡，其盗版率为 20%。在这些被调查的国家中，2007 年的盗版率的中间值为 61%，这就意味着世界上有一半的国家的软件盗版率达到了 61%或者更高。2011 年软件盗版造成的经济损失已经攀升至 63 亿美元。所有这些数据表明，软件盗版形势非常严峻，软件产品的版权保护以及打击盗版问题已经变得非常必要(见图 1-1)。

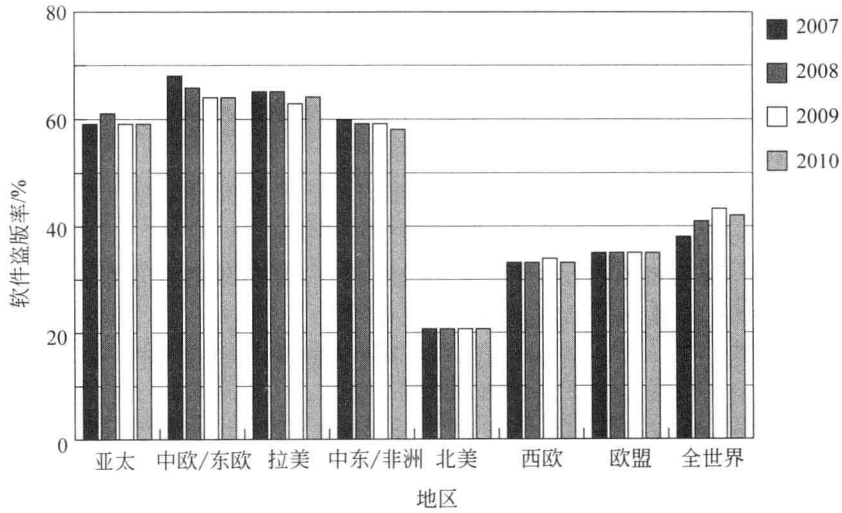

图 1-1　全球各区域软件盗版概率图

完善的法律法规和有效的技术保护是解决盗版问题的保障。但是，由于软件兼有作品和技术的双重特性，在软件保护的过程中形成了许多特殊的地方：首先，软件产品的复制成

本低，且不易被察觉和惩罚；其次，软件产品的种类复杂，使用的技术也各不相同，因而保护的形式也各不相同；再者，软件法律保护的形式有版权、专利、商业秘密和商标等，这些形式在保护的过程中既各有侧重也存在重叠，因而导致了软件在法律保护形式上有一定的困难性。显然，就目前人们对软件的认识来看，要想设计出一种完全意义上的软件保护方案，既是不可能的，也是与软件自身特点相冲突的。但是，随着软件产品复杂性的增加，软件保护技术的逐渐成熟以及相关法律法规的健全，还是可以在很大程度上保障软件的分发和运行安全的，其中以设计和提出切实可行的保护技术显得尤为重要，也具有深远的现实意义。软件水印、软件混淆和软件防篡改技术正是这方面的一种尝试，是为解决软件知识产权保护问题而发展起来的一个新兴学科。

Java 语言由于其平台无关性得到了业界内的广泛认可，并在关键领域内得到了应用。但是，Java 程序却面临着严重的未授权使用、反编译和恶意篡改等问题，这是由于 Java 程序主要以 Class 文件形式存在，这种 Class 文件的格式已在《JavaTM 虚拟机规范》中公开，使得攻击者很容易重用或破解 Java 程序，甚至反编译 Java 程序为源代码进行使用。因此，Java 程序的版权保护和内容安全性问题显得更为严重。

为了应对盗版、逆向工程和篡改程序这 3 种对软件的破坏方式，分别出现了软件水印、软件混淆和防篡改技术 3 种软件保护新兴技术。软件水印可以通过在软件产品中嵌入水印信息来保护软件的版权、追踪盗版的根源等做法，以弥补加密方案不能对解密后的软件提供进一步保护的不足，也能弥补审计方案不能追踪盗版根源的不足，为软件保护提供了一种全新的思路；软件混淆技术可以对程序进行变换，让程序更加难以理解，增加逆向工程的难度；防篡改技术可以检测到程序是否被篡改，如果发现程序被篡改，就发出响应或停止程序运行。

1.2 软件保护场景

软件安全问题的日益严重，引起了学术界对软件保护及其相关技术的重视。软件安全领域通常针对两类攻击主体对象（不可信的软件程序和不可信的软件宿主）进行研究。

对于第一类攻击，由于运行环境是可控的，因此对攻击的抵御相对容易；对于第二类攻击主体，攻击者通常使用以下 3 种方式攻击软件：

（1）获取非授权访问。攻击者通过某种攻击方式使得软件中的访问控制机制失效，重新分发非法的软件副本从而获利。这是常见的软件盗版问题。

（2）逆向工程。攻击者通过反编译/反汇编技术，获得软件的全部或部分源代码，从而获取关键信息，如核心算法、秘密信息等，为自己所使用。

（3）破坏代码完整性。攻击者向软件代码中嵌入恶意代码或修改、删除部分代码以达到自己的目的。

在一次攻击事件中，攻击方式并不是唯一的。例如，为了获取软件的非授权访问，通常要结合逆向工程和破坏代码完整性两种攻击方式。目前，在整个软件保护研究领域中，抵御上述 3 种攻击方式的应对手段分别是软件水印、软件混淆和软件防篡改。图 1-2 描绘了这 3 种技术手段：

- 软件水印。在图 1-2(a)中，Alice 在发布软件之前利用密钥 key 向软件中嵌入水印 W。拥有密钥 key 的 Tom 能够提取水印 W 以证明软件为 Alice 所拥有。这样就达

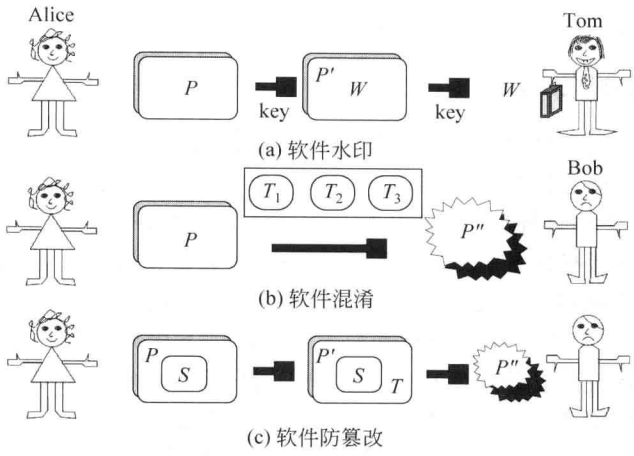

图 1-2　应对 3 类宿主攻击的防御机制

到了保护版权的目的。

- 软件混淆。在图 1-2(b)中，Alice 在发布软件之前，利用一系列混淆变换技术，将其代码转换为功能、语义相同但更难为攻击者 Bob 阅读理解的软件，以阻碍 Bob 对软件实施逆向工程。
- 软件防篡改。在图 1-2(c)中，Alice 在发布软件之前向软件中增加防篡改模块以保护软件的秘密信息 S。若攻击者 Bob 篡改软件中的秘密信息 S，则他无法正常运行软件。

由于数字水印技术是近年来国际学术界的一个前沿研究领域，处于迅速发展阶段，而软件水印又是其中的一个重要组成部分，因此，掌握其发展方向对软件水印的研究有着重要的指导意义。今后软件水印技术的研究和其他多媒体水印技术一样，也应该侧重于完善理论，提高水印算法的鲁棒性，建立相关标准。而且软件保护方式的设计应在一开始就作为软件开发的一部分来考虑，列入开发计划和开发成本中，并在保护强度、成本和易用性之间进行折中考虑，选择一个合适的平衡点。

1.3　软件攻击模型

在讨论软件水印的攻击模型时，必须对攻击者的目的、能力和手段有一个充分的认识，然后才能对此提出一个切实可行的阻止措施和保护方案。

1.3.1　攻击者的能力

对于任何一个软件水印方案，必须充分估计攻击者的一切能力。通常，一个攻击者具备下述能力：

- 任意查看软件中的代码和数据；
- 任意修改软件中的代码和数据；
- 让软件在任何点开始运行或停止运行。

当攻击者确定自己的攻击目标之后，会不惜一切代价，施用一切方式来实现自己的攻击目标，通常，这些攻击方式包括针对软件保护(例如软件水印、软件胎记和防篡改等)的通用

攻击方式和针对软件水印的特定攻击方式。

1.3.2 攻击者的目标

软件水印可以应用于软件的版权保护、跟踪盗版、完整性验证以及许可控制等方面，因此攻击者的目标也会随着软件水印的不同应用功能而不同。具体来说，当软件水印应用于版权保护时，水印应该是鲁棒的，这时攻击者的目标就是破坏水印或者嵌入一个盗版水印，使得原水印不再具有版权保护作用，从而能使自己免于相关法律责任的追究。当软件水印应用于跟踪盗版或者跟踪交易记录时，水印也应该是鲁棒的，攻击者的目标就是破坏软件中的指纹水印，或者将其中的指纹水印修改为一个与自己不相关的指纹水印，从而能够自由地复制、分发修改后的软件，即使盗版行为被发现，也不必担心自己会被怀疑和定位。当水印应用于软件的完整性验证或者许可控制时，水印应该是脆弱的，而攻击者的目的往往不是破坏水印，而是破坏完整性验证条件和许可验证条件，使得软件还能正常使用。

1.3.3 针对软件保护的通用攻击方式

根据攻击的自动化程度，可以将针对软件保护的通用攻击方式分为人工攻击和自动攻击。前者通过人工分析，攻击者可以找出与程序运行结果不相关的代码和数据，然后去除这部分代码和数据以达到破坏软件保护的目的。后者则借助工具，对程序进行代码混淆、代码优化等语义保留攻击，以达到破坏软件保护的目的。

当仅仅通过简单的攻击不能破坏软件保护的时候，攻击者会从程序中采集一些信息，以帮助自己理解程序和实施更深入的攻击。根据信息采集的方式，可以将这些信息分为静态信息和动态信息。静态信息的采集仅需要检查程序的代码和数据本身，进而对程序的可能执行行为进行适当地推理，而不需要运行程序。动态信息的采集需要运行程序本身，进而对程序的一个或者多个实例的运行行为进行观察。静态信息和动态信息是彼此互补的，可以首先采集一种信息，在此基础上再去采集另一种信息，如此反复迭代地进行信息采集，采集的次数越多，对程序的理解越深。类似地，根据采集信息的类型，可以将软件保护的攻击方式分为静态攻击、动态攻击和混合攻击。静态攻击是仅仅基于静态信息的攻击方式，动态攻击是仅仅基于动态信息的攻击方式，而混合攻击是综合运用静态信息和动态信息的攻击方式。

1.3.4 针对软件水印的特定攻击方式

如图 1-3 所示，攻击者针对软件水印的特定攻击方式主要包括 4 种：添加攻击、变形攻击、去除攻击和合谋攻击。

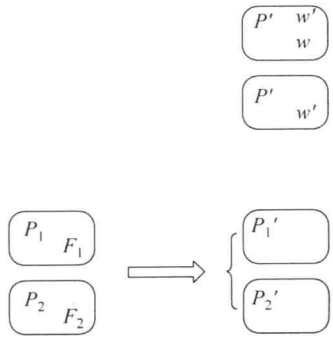

图 1-3 软件水印攻击模型

（1）添加攻击：往应用程序中添加新的水印，试图破坏原水印，或者使得原水印在嵌入时间上晚于新嵌入的水印。

（2）变形攻击：通过混淆、优化等手段对程序进行语义保留攻击，试图使水印发生变形或者不可提取，从而不再具有保护作用。

（3）去除攻击：分析程序，确定水印的隐藏位置，然后删除水印；一种典型的去除攻击就是反编译-再编译。

（4）合谋攻击：通过比较两个含有不同指纹水印的程序，找出指纹水印的隐藏位置，然后破坏掉水印。

1.4　研究现状

随着软件产业的迅速发展，软件产品的版权保护已成为一个十分重要的问题，越来越受到人们的关注。研究与分析软件版权保护的相关内容，对于保护计算机软件著作权人的权益，鼓励计算机软件的开发与应用，促进软件产业和国民经济信息化的发展有着重要的意义。其中，Java Class 文件的平台无关性使得它在网络上广泛传播，因此对 Class 文件的保护显得尤为关键。

软件水印是程序分析、软件工程、密码学和算法设计等学科的交叉研究领域。目前，国外主要有新西兰的奥克兰大学、美国的亚利桑那大学、威斯康星大学、日本的奈良工业科技大学和大阪大学等在做这方面的研究工作，并分别取得了一定的成果，其中以奥克兰大学和亚利桑那大学最为领先。国内的研究则起步较晚，目前清华大学、北京邮电大学、浙江大学、同济大学、湖南工业大学和西北工业大学等院校均已开展了相关的研究工作。

在 1996 年，美国公示了第一个软件水印专利，这标志着软件水印技术的正式诞生，随后，软件水印技术吸引了越来越多的学者进行深入的研究，目前已发表了 10 个软件水印算法，其中包括 7 个静态软件水印算法和 3 个动态软件水印算法，我们将在 3.5 节详细地介绍这 10 个算法。

在软件水印系统和工具方面，奥克兰大学和亚利桑那大学两所高校合作开发的 SandMark[13] 系统是软件保护领域里最为重要的研究工具，其中包括了 13 个静态软件水印算法（由 6 个基本算法和已发表的 7 个静态算法组成）和 3 个动态软件水印算法；威斯康星（麦迪逊）大学针对动态软件水印进行了深入的研究并开发了 UWStego 系统；日本的奈良工业科技大学也在软件水印领域开展了积极的探索，并开发了 Jmark 系统。

软件水印可以针对源代码、中间代码和本地代码进行研究。不管针对哪一种代码进行研究，其原理和技术是通用的，即主要由程序分析技术和代码操纵技术组成。目前，软件水印的研究主要针对中间代码，几乎所有的工作都集中在 Java 字节码上，原因有三：首先，软件产品的作者本人往往不愿意公开源代码，因为针对源代码的软件水印研究需求显得很少；其次，对于本地代码，由于其对具体的平台具有很强的依赖性，分析和操纵本地代码并不是很方便，这也导致了针对本地代码的软件水印研究很少；最后，由于 Java 字节码文件格式已公开，很多开源组织都提供了效率很高且免费的字节码操纵和分析工具，例如 Apache 提供的 BCEL 和 ObjectWeb 提供的 ASM 等，所有这些工具使得分析和操纵 Java 字节码变得非常容易，因此软件水印也主要针对 Java 字节码进行研究。

本书第 3 章介绍了软件水印这种新兴软件版权保护技术，分析了软件水印的现状、分类、攻击方法以及已有的各种算法。简要说明了清华大学信息系统与工程研究所现有的针对 Class 文件水印算法的 TRUP 平台。在广泛了解软件水印的算法基础上，重点阐述了其中若干典型的水印算法，并进行了算法分析评价。

1.5 软件保护方案

1.5.1 基于审计的保护方案

一些组织，例如商业软件联盟(BSA)可以通过审计来核实一个企业或者组织是否在使用盗版软件。这种审计主要是指检查计算机系统中与软件相关的材料清单。这些材料包括：

- 安装的所有媒介；
- 所有的手册和参考文档；
- 与许可证相关的文件；
- 所有能证明软件合法性的文件，例如发票。

基于审计的保护方案能够核查出一个企业是否在使用盗版软件，也能帮助企业核查出其内部员工是否存在非法盗用行为。但是，其缺点在于不能找出盗版的根源所在。

1.5.2 基于硬件的保护方案

一些具有特殊用途的硬件可以用于软件保护，例如证明版权、提供安全的数据存储以及提供安全的运行环境等。常见的保护手段包括加密狗、防篡改 CPU 以及智能卡 3 种。

加密狗(dongle)也叫软件狗或软件保护器，是伴随软件分发的一个硬件装置，其中包含了一些厂家定制的专用集成电路。通常，它被附加在计算机的 I/O 端口(例如 USB 接口)上。在运行过程中，软件会定时地检查加密狗，如果检查失败或者检查结果不正确，那么软件就会做出相关的反应。

防篡改 CPU 能通过提供安全的数据存储和运行环境来保护软件。盗版软件不能访问那些在安全环境中运行的软件，因而攻击者就不能观察软件的运行行为。

智能卡由一个或多个带有微处理器和存储器的集成电路芯片组成，并封装成便于携带的卡片。它具有暂时或永久的数据存储能力，其内容可供外部读取或内部处理或判断之用，同时还具有逻辑处理功能，用于识别和响应外部提供的信息和芯片本身判定路线和指令执行的逻辑功能。一个典型的应用是智能卡可以用来存储广播电视系统中频道识别的密钥。

基于硬件的保护方案有着很多缺点。从用户角度来说，硬件的部署以及软件的使用往往不太方便，支撑软件运行的硬件设施升级困难等等；从卖主的角度来说，硬件的开发代价比较大，从而导致软件的成本比较大、价格比较昂贵等等。

1.5.3 基于软件的保护方案

目前，基于软件的保护技术主要包括软件加密、软件水印、软件胎记、软件混淆和防篡改等几种。

软件加密一般是用户在发送信息前，先调用信息安全模块对信息进行加密，然后发送；到达接收方后，由用户使用相应的解密软件进行解密并还原。

软件水印技术是指不被感知地向软件中嵌入一些标识信息，这些标识信息可以用于证明软件版权、跟踪盗版、完整性验证和控制软件的使用等。

软件胎记技术是指在不修改软件的前提下，从中提取出特征信息，用于标识软件的版权。如果两个程序的胎记很相似，那么其中一个程序是另一个程序的副本的可能性就很大。

软件混淆（也称代码混淆）是指不改变程序原有功能的前提下，对程序进行语义保留变换，使程序变得晦涩难懂。代码混淆通常用于阻止恶意的反向工程。

防篡改是一种保护软件不被恶意修改的技术，它包括检测和响应两个过程。经防篡改保护后的程序，一旦检测到程序被篡改了，就会激活响应模块，使得程序以某种隐蔽的方式不能正常运行。

与基于审计的保护技术相比，基于软件的保护技术有着更全面的功能，例如跟踪盗版。与基于硬件的保护技术相比，基于软件的保护技术有着开发代价小、便于部署、便于升级等优点。本书的重点研究方向就是基于软件的保护技术。

1.6　关于软件保护的专利及法律条文

随着网络技术的发展，计算机软件的法律保护开始逐渐显现出必要性。早在 1965 年，前联邦德国的学者已经提出以法律保护计算机软件的核心——计算机程序。从 1980 年开始，计算机程序受到版权法保护，才逐渐在各国得到普遍确认；同时，针对计算机软件的法律保护问题不断展开讨论，随之而来的是各国的相关立法、司法以及计算机技术的发展和不断深化。下面简要介绍美国、欧盟、日本以及中国的软件保护历程。

1.6.1　美国

美国计算机软件保护相对于其他国家起步比较早，并且更为健全。其发展经历了一个曲折而又反复的过程，从最初的反对软件专利保护，到后来的积极推动全球软件版权保护，以及如今的扩大软件专利保护。

20 世纪 60 年代初期开始，美国政府着手修订版权法并试图用版权制度来保护程序，先后向两院提交了十几个法律草案，但都未能通过。直到 1974 年年底才通过一项议案设立了“版权作品新技术应用国家委员会”（简称 CONTU），该委员会就软件保护等问题进行了大量的调查研究工作，对美国版权立法产生了很大的影响。1980 年 12 月，美国国会修订了 1976 年版权法第 101 条与第 117 条，正式把计算机程序列入著作权法的保护范围。美国著作权法中对于可以受到保护的计算机程序只在原则上说了一句话：“旨在直接用于计算机以取得一定结果的一组语句或指令”（set of statements or instructions to be used directly in a computer in order to bring about certain result）。

随着计算机软件技术的发展，1994 年以来，美国法院对计算机软件的可专利性采取了更加宽松的态度，1996 年 USPTO（美国专利商标局）发布的《与计算机有关的发明的审查指南》中规定：只要计算机软件发明申请在技术领域中有实用性，就是法定的可获得专利保护的主题。1998 年以后，商业方法软件也可在美国获得专利保护。2000 年 3 月 29 日，USPTO 发表了《关于自动财务/管理数据处理方法（商业方法）的白皮书》，明确了数据库和商业方法可以作为软件专利申请的主题。

CONTU 认为计算机程序版权保护应遵循“思想/表达二分法”，不采用特殊的审查标准，而应该依据不同案例的不同情况加以判断，这表明美国坚持用传统版权法来保护计算机

软件。由于数字网络技术对版权法影响很大，1998 年 10 月美国又颁布了《数字化千年版权法案》对原来的版权法进行了适应数字网络环境的修订。

1.6.2 欧盟

《欧洲专利公约》(EPC) 规定计算机程序不属于发明，但又指出只有有关计算机程序本身的发明才被排除在可专利性之外。如果要就计算机软件申请发明专利，则必须是计算机程序加上别的东西，即与计算机程序相关的发明。判断计算机软件的可专利性问题时，常常要求有超出载体的东西存在，这就是欧洲软件专利审查中的一个核心概念——技术效果。此外，发明必须作为整体考虑，如果发明同时利用了技术手段和非技术手段(如算法)，非技术手段的运用并不能转移整个发明的技术特性。2000 年后欧洲软件专利的审查标准已放宽到开始为软件产品和含有商业方法的装置提供专利保护。欧洲对计算机软件的可专利性问题同样经历了由不认可到有条件认同，进而到今天态度更为宽松的转变过程。

欧洲各国在立法上否定了软件专利，在专利法中规定了不受保护的范围，例如，1997 年的英国专利法第一条(2)款 C 项、1980 年联邦德国专利法第一条(2)款 3 项，以及法国现行专利法都规定计算机软件不在受保护范围之内。他们认为，软件在性质上只是抽象的算法，属于科学原理，不符合专利法的保护条件。因此，欧洲各国对计算机软件的法律保护大都借助于版权法。

1.6.3 日本

日本关于计算机程序专利审查标准的演变是由最初确认软件的可专利性，到从整体看待与软件有关的发明，进而发展到将专利保护延及载有计算机程序的磁盘或光盘，直至近年对与商业有关的发明的审查，其审查标准在逐步放宽。但其审查标准的基点，即法定的发明的定义，却一直未曾动摇，强调“应用自然法则”的标准一直在执行着。对与商业有关的软件发明，只要满足发明的定义条件，就视为可专利性的主题。其对软件可专利性的审查标准的演变紧随美国之后。

日本在 1985 年立法将计算机软件法律保护纳入著作权法框架。其中软件保护条款是在 1985 年 6 月 14 日修订时首次出现，1986 年 1 月 1 日起实施，日本现行著作权法于 2000 年 5 月 8 日修订，2001 年 1 月 1 日起实施。日本著作权法第 113 条共有 5 款，其中第二款规定：“在商业行为中，在计算机上使用通过侵犯程序作品著作权而制作的复制品的行为，视为侵犯该项著作权的行为，只要在获得这些复制品时，使用者知道上述侵权。”因此，日本规定明知是侵权软件而在商业行为中或在业务上使用的视为侵权。

1.6.4 中国

目前，中国有关计算机软件著作权保护的法律、法规和司法解释有《中华人民共和国著作权法》、《中华人民共和国著作权法实施条例》、《计算机软件保护条例》、《中美两国政府关于保护知识产权的谅解备忘录》、《计算机软件著作权登记办法》、《实施国际著作权条约的规定》、《著作权行政处罚实施办法》、《中华人民共和国刑法》、《最高人民法院关于审理非法出版物刑事案件具体应用法律若干问题的解释》和《最高人民法院关于审理著作权民事纠纷案