高等院校信息管理与信息系统专业参考教材

信息系统的安全与保密

贾　晶　陈　元　王丽娜　编著

清华大学出版社

(京)新登字 158 号

内 容 简 介

本书全面系统地论述了信息系统安全保密的基础理论及实用技术。全书共分四部分,第一部分概述了计算机信息系统安全保密的重要性及研究内容;第二部分介绍了密码学的基础理论知识,讲述了传统密钥体制和公开密钥体制;第三部分详细讲述了信息系统安全保密的实用技术,并且重点强调了网络环境下防火墙安全措施的应用;第四部分通过对计算机病毒危害及症状的分析,论述了防止病毒的常用方法。本书内容丰富,覆盖面广,适用于大专院校信息管理与信息系统、计算机应用等相关专业学生阅读,而且对从事计算机信息系统安全工作的技术人员也有极大的帮助。

图书在版编目(CIP)数据

信息系统的安全与保密/ 贾晶主编 .—北京: 清华大学出版社,1998 .12
高等院校信息管理与信息系统专业参考教材
ISBN 7-302-03213-0

Ⅰ .信… Ⅱ .贾… Ⅲ .①信息系统-安全技术-高等学校-教材 ②电子计算机-保密技术-高等学校-教材 Ⅳ .G202

中国版本图书馆 CIP 数据核字(98)第 33076 号

出版者: 清华大学出版社(北京清华大学校内,邮编 100084)
http: // www .tup .tsinghua .edu .cn
印刷者: 印刷厂
发行者: 新华书店总店北京发行所
开 本: 787×1092 1/ 16 印张: 10 字数: 233 千字
版 次: 1999 年 1 月第 1 版 1999 年 1 月第 1 次印刷
书 号: ISBN 7-302-03213- 0/ TP·1717
印 数: 00001～10000
定 价: 0.00 元

序　　言

全国计算机基础教育研究会财经信息专业委员会,从 20 世纪 90 年代初期以来,对于信息管理与信息系统专业的学科建设和教材建设,投入了大量的精力。在清华大学出版社的大力支持下,专业委员会组织全国有关院校的同行们,陆续出版了一批为信息专业迫切需要且具有一定特色的教材,这就是几年来已经陆续出版的“信息管理与信息系统专业系列教材”。1997 年夏天,在烟台举行专业委员会的学术年会上,来自全国各地教学第一线的同行们,进一步讨论了信息管理与信息系统专业的学科建设。针对该专业内容新、跨度大、变化大的特点,大家一致认为,有必要再组织一套参考书,以满足这个专业本科高年级选修课和研究生课程的需要。这就是目前这一套“信息管理与信息系统专业参考教材”的由来。

最近由教育部正式颁布实行的本科专业目录中,信息管理被列为管理门类之下的一个二级学科。这表明,经过 20 年的成长与发展,随着信息化建设的深入,信息管理已得到社会各界的认可,成为管理学科建设与现代化管理人才培养的一个不可缺少的组成部分。按照教育部的本科专业目录,原先分散在各领域中的经济信息管理、管理信息系统、科技信息管理等,均归入“信息管理”名下,成为一个覆盖面更宽的统一专业。对于从事该领域工作的教师来说,是给予了充分的肯定和大力的支持,同时也意味着面临着新的、要求更高的学科建设任务。本专业委员会的全体同志决心以面向 21 世纪的新标准,进一步创新和探索,为信息管理与信息系统专业的进一步发展努力奋斗。

本套书与前几年出版的“信息管理与信息系统专业系列教材”不同,它不属于基本的核心课程,而是面向本科高年级的选修课和研究生的课程。按照教育部专业调整的精神,专业设置不宜过窄过细,而应当宽口径、厚基础,给学校、教师和学生以更大的发展余地。体现在课程设置中,就意味着应当增加选修课程,使学科能够在宽口径的专业设置中办出自己的特色,使学生能够在厚基础的前提下有更多的选择。而要做到这一点,就需要提供一大批供选择的课程和教材,本套书就是为此目的而组织编写的。显然,对于信息管理与信息系统这样一个内容新、发展快、综合性强的专业,这方面的需求无疑将更为迫切。

每一个专业都有自身最核心的一些内容,它包括从事本专业工作所必需的基本概念、基础知识、基本技能、基本素质,即平时所谓的“看家本领”。然而,在新技术革命的浪潮冲击下,知识与技术的更新速度大大加快,各领域知识互相渗透,综合运用的趋势不断加强,指望在大学四年中准备好一生工作所需要的知识,是不可能的。同样,囿于专业分工,只靠某一狭窄的专业领域中的知识和技能,将很难适应未来多变的社会需求。因此,一方面,拓宽视野、了解和掌握相关学科的知识对于提高素质和适应能力十分必要;另一方面,及时掌握新的技术生长点,了解学科和技术的最新发展方向,对于学生发展的后劲也是必不可少的。本套书的第一批书目正是根据以上两方面的思路选定的。

信息管理与信息系统也是信息科学的一个部分,它以现代信息技术为手段和基础,同时

又与信息经济学、信息社会学、信息法学、系统科学密切相关，一个称职的、高水平的信息管理人员，对于这些知识都应当有一定深度的了解。对于信息管理领域和信息技术领域的一些新发展，如电子商务、数据挖掘等，信息管理与信息系统专业的学生，特别是有兴趣向这些方面发展的学生，也是应当有所了解的。毫无疑问，这些方面的具体内容发展变化是很快的，第一批选题不可能覆盖所有应当考虑的范围。目前这套书只是开头，以后必然要不断地增加、补充。同时，已经出版的几种书，也将随着技术和社会的发展，不断修订和补充，以便切实为各院校从事信息管理与信息系统专业建设的同行们提供帮助。

在清华大学出版社的大力支持下，本套参考教材第一批已经陆续问世，这是有关院校与老师共同努力的初步成果。由于这项工作是尝试性的，能否实现酝酿时的初衷，还需要实践的检验。因此，我们迫切希望得到各院校以及社会各界的批评指正，从选题范围到具体内容，都希望能够得到中肯的批评和建议。我们特别欢迎在信息化建设第一线的同志们，从信息化人才培养的实践需要出发，对于本套书的方针和内容提出意见，并进一步参与本套书的编写工作。

全国计算机基础教育研究会
财经信息管理专业委员会
信息管理与信息系统专业参考教材编委会
主任：陈禹，副主任：张基温
1998 年 7 月

前　言

当今社会是信息化社会,电子计算机和通信网络已经广泛地应用于社会的各个领域,利用这些先进技术建立起来的信息系统正改善着人们的生活和工作方式。然而,在我们享受着众多信息及信息系统带来的巨大方便的同时,也时时受到来自各方面对信息系统安全的威胁。据美国 FBI 统计,每年因信息和网络安全问题所造成的损失高达 75 亿美元,法国为 80 亿法郎。此外,计算机病毒也是一个众人皆知的恼人问题,会对计算机信息系统及网络造成严重的破坏,因此计算机信息系统的安全保密成为了迫切需要解决的问题。本书就是为高等院校相关专业学生和从事信息处理工作的同志学习及研究信息系统安全保密技术而编写的。

本书不仅包括信息安全、密码学方面的基础理论知识,而且重点讲述了信息系统安全保密的实用技术。该书内容丰富、材料详实、重点突出、可读性强,它既可以作为系统学习信息安全知识的教材,也可以作为信息工作人员解决安全问题的实用手册。

全书共分 7 章,第 1 章概述了信息系统安全的重要性及其主要的研究内容;第 2 章介绍了密码学的预备知识:数论、信息论和计算复杂性理论;第 3 章介绍了传统密码体制及数据加密标准;第 4 章介绍了几种常见的公钥体制:RSA 体制、背包体制、ELGamal 体制、概率加密体制;第 5 章首先讲述了操作系统和数据库的安全保密问题,然后讨论了数字签名、智能卡和电子数据传输 EDI 的安全技术;第 6 章结合实例介绍了网络安全策略——防火墙技术与秘密邮件技术;第 7 章分析了计算机病毒的危害及症状,介绍了检测和防治病毒的实用方法。本书的第 1、第 4、第 7 章由贾晶编写,第 2、第 3 章由陈元编写,第 5、第 6 章由王丽娜、贾晶共同编写。贾晶负责全书的总体规划与内容的组织,并且对全书进行了修改和定稿。

本书在编写过程中得到了中国人民大学陈禹教授、山西财经大学张基温教授、天津财经学院王晓堤教授和清华大学出版社的大力支持与帮助,在此表示衷心感谢。此外,为了编著本书我们参考和吸收了国内外许多同行学者的研究成果,很多朋友也为此书付出了辛勤的劳动,在此一并致谢。

由于我们水平有限,加上时间紧促,书中错误在所难免,欢迎批评指正。

作者

1998 年 5 月

目　　录

第 1 章　信息系统安全概述

当今社会是信息化社会,电子计算机和通信网络已经广泛的应用于社会的各个领域,以此为基础建立起来的各种信息系统,给人们的生活、工作带来了巨大变革。大型信息系统将众多的计算机和智能化设备连在一个四通八达的通信网络中,共享丰富的数据库信息和计算机资源,存储大量的数据文件,完成异地之间的数据交换与通信。信息系统的应用,加速了社会自动化的进程,减轻了日常繁杂的重复劳动,同时也提高了生产率,创造了经济效益。

然而信息系统越是重要,它就越容易受到攻击,攻击者以求从中获取非法的利益,因此,信息系统的安全保密成为迫切需要解决的问题。目前,在信息系统的开发设计过程中,安全性能被放在首要的位置,成为信息系统生存的关键。本章概述了信息系统安全的重要性及其主要的研究内容,介绍了信息系统安全条例和国际信息安全的著名机构。

1.1　信息系统安全的重要意义

信息、材料和能源是人类社会赖以生存和发展的基础。在现代信息化社会里,人类的一切活动都离不开对信息的获取与处理,信息作为一种无形资产已经成为人们最宝贵的财富。信息系统是各种方法、过程、技术按一定规律构成的一个有机整体,它是信息采集、存储、加工、分析和传输的工具,为了使敏感信息安全可靠,必须确保信息系统的安全可靠。本节将重点讨论信息系统的安全性问题。

1. 信息系统的概念

信息系统是复杂系统中的小系统,它的作用是沟通各个子系统使整体系统协调一致。社会组织的类型和功能各不相同,但它们都有自己一定形式的信息系统,而且信息系统功能的好坏直接影响全局组织的效益,因此可以说信息系统是整个系统的中枢。信息系统的功能包括以下几个方面。

1) 信息的采集

这是信息系统最基本的功能,它将分散在各部门、各地方的相关信息收集起来,记录其数据,然后整理成为信息系统所需要的格式。

2) 信息的加工

信息系统对信息进行加工处理的具体含义是:排序、分类、归纳、检索、统计、模拟、预测以及各种数学运算。所有这些工作,都依靠规模大小不同的计算机来完成。

3) 信息的存储

信息系统经过上述两个步骤后,形成了对管理决策有价值的信息,由信息系统完成对

这些有用信息的存储保管。若信息过于庞大,就必须有良好的组织结构和先进的存储技术。通常,信息的存储分为逻辑组织和物理存储两个方面,前者依赖合理的数据结构,后者把信息存储在合适的介质上。

4) 信息的检索

检索查询是信息系统重要的功能,存储在系统内的信息要便于查询,满足各方面用户的需求,通常信息检索需要用到数据库的技术与方法。

5) 信息的传输

信息系统中的信息必须具有相互交换的功能,各部门之间要完成通信工作,一定要解决好信息的传输问题,以便使信息准确迅速地送往使用者手中。

信息系统的设计需要针对不同的管理层次,由 R .N .Anthory 提出的三级管理模型,将管理过程划分为三个层次:战略计划、管理控制、操作运行,如图 1 .1 所示。

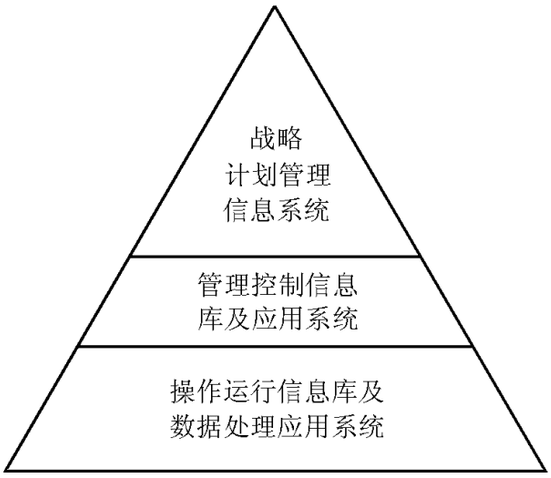


图 1.1　信息系统的层次模型

2. 信息系统受到的威胁

由于信息系统是以计算机和数据通信网络为基础的应用管理系统,因而它是一个开放式的互联网络系统,如果不采取安全保密措施,与网络系统连接的任何终端用户,都可以进入和访问网络中的资源。目前,管理信息系统已经被用于金融、贸易、商业、企业各个部门,它在给人们带来极大方便的同时,也为那些不法分子利用计算机信息系统进行经济犯罪提供了可能。据不完全统计,每年因利用计算机系统进行犯罪所造成的经济损失高达上千亿美元。在我国,利用管理和决策信息系统从事经济活动起步较晚,但各种计算机犯罪活动已时有报道,直接影响了信息系统的普及使用。归纳起来,信息系统所面临的威胁分为以下几类。

1) 通信过程中的威胁

信息系统的用户,在进行信息通信的过程中,常常受到两方面的攻击:一是主动攻击,攻击者通过网络线路将虚假信息或计算机病毒,卷入信息系统内部,破坏信息的真实性与完整性,造成系统无法正常运行,严重的甚至使系统处于瘫痪;二是被动攻击,攻击者非法窃取通信线路中的信息,使信息机密性遭到破坏,信息泄漏而无法察觉,给用户带来巨大的损失。

2) 存储过程中的威胁

存储于计算机系统中的信息,易于受到与通信线路同样的威胁。非法用户在获取系统访问控制权后,浏览存储介质上的机密数据或专利软件,并且对有价值的信息进行统计分析,推断出所需的数据,这样就使信息的保密性、真实性、完整性遭到破坏。

3) 加工处理中的威胁

信息系统一般都具有对信息进行加工分析的处理功能,而信息在进行处理过程中,通

常都是以源码出现,加密保护对处理中的信息不起作用。因此,在这个期间有意攻击和意外操作都极易使系统遭受破坏,造成损失。

除此之外,信息系统还会因为计算机硬件的缺陷、软件的脆弱、电磁辐射和客观环境等原因造成损害,威胁计算机信息系统的安全。

3. 对信息系统攻击的主要手段

信息系统在运行过程中,往往受到上述各种威胁和攻击,非法者对信息系统的破坏主要采取如下手段:

· 冒充　这是最常见的破坏方式。信息系统的非法用户伪装成合法的用户,对系统进行非法的访问。冒充授权者,发送和接收信息,造成信息的泄露与丢失。

· 篡改　通信网络中的信息在没有监控的情况下,都可能被篡改,即将信息的标签、内容、属性、接收者和始发者进行修改,以取代原信息,造成信息失真。

· 窃收　信息盗窃可以有多种途径,在通信线路中,通过电磁辐射侦截线路中的信息;在信息存储和信息处理过程中,通过冒充、非法访问,达到窃取信息的目的。

· 重放　将窃取的信息,重新修改或排序后,在适当的时机重放出来,从而造成信息的重复和混乱。

· 推断　这也是在窃取基础之上的一种破坏活动,它的目的不是窃取原信息,而是将窃取到的信息进行统计分析,了解信息流大小的变化、信息交换频繁程度,再结合其他方面的信息,推断出有价值的内容。

· 病毒　病毒对计算机系统的危害是共所周知的,目前已经发现的计算机病毒达几千种,它直接威胁着计算机的系统和数据文件,破坏信息系统的正常运行,甚至造成整个系统的瘫痪。

总之,对信息系统的攻击手段多种多样,我们必须学会识别这些破坏手段,以便采取技术策略和法律制约两方面的努力,确保信息系统的安全。

1.2　信息安全技术的研究内容

一个信息系统是否可行,在很大程度上取决于此系统安全性能的好坏,要看其是否能够实现用户提出的安全要求。然而要确保信息系统的安全可靠,离不开信息安全技术的应用。本节重点介绍信息安全技术研究的内容。

1. 信息安全技术的含义

信息安全技术是一门综合的学科,它涉及信息论、计算机科学和密码学多方面的知识,它研究计算机系统和通信网络内信息的保护方法,以实现系统内信息的安全、保密、真实、完整。信息安全技术具体包括如下几方面的含义。

1) 保密性(confidentiality)

信息或数据经过加密变换后,将明文变成密文形式,表面上无法识别,只有那些经过授权的合法用户,掌握秘密密钥,才能通过解密算法将密文还原成明文。而未授权的用户

因为不知道秘密密钥,而无法获得原明文的信息,起到对信息保密的作用。

2) 完整性(integrity)

完整性指的是将信息或数据附加上特定的信息块,系统可以用这个信息块检验数据信息的完整性,特点是信息块的内容通常是原信息或数据的函数。只有那些经过授权的用户,才允许对数据或信息进行增加、删除和修改。而未经过授权的用户,只要对数据或信息进行改动就立刻会被发现,同时使系统自动采取保护措施。

3) 可用性(availability)

可用性指的是安全系统能够对用户授权,提供其某些服务,即经过授权的用户可以得到系统资源,并且享受到系统提供的服务。防止非法抵制或拒绝对系统资源或系统服务的访问和利用,增强系统的效用。

4) 真实性(authenticity)

真实性指的是防止系统内的信息感染病毒。由于计算机病毒的泛滥,已很难保证计算机系统内的信息不被病毒侵害,因此信息安全技术必须包括反病毒技术,采用人工方法和高效反病毒软件,随时监测计算机系统内部和数据文件是否感染病毒,一旦发现应及时清除掉,以确保信息的真实可靠。

随着信息系统的进一步发展,新的安全性要求仍在不断地提出,这对信息安全技术起到了促进推动作用,相信不久的将来会有更新、更实用的安全方法开发出来。

2. 信息系统安全模型

信息系统安全是一项复杂的系统工程,它的实现不仅是纯粹的技术方面问题,而且还需要法律、管理、社会因素的配合。因此信息系统安全模型是一个层次结构,如图 1.2 所示:

图 1.2 信息系统安全层次模型

从图中可以看出各层之间相互依赖,下层向上层提供支持,上层依赖于下层的完善,最终实现数据信息的安全。

1) 第一层是法律制度与道德规范

日益严重的计算机犯罪,迫使各国尽快制定出严密的法律、政策,从根本上改变知识犯罪无法可依的局面。以全新的概念和要求,规范和制约人们的思想与行为,将信息安全

纳入规范化、法制化和科学化的轨道。

2) 第二层是管理制度的建立与实施

安全管理制度的建立与实施,是实现计算机信息系统安全的重要保证。它包括:安全管理人员的教育培训、制度的落实、职责的检查等方面的内容。下面举例说明。

例: 天津财经学院计算机信息系统安全管理条例如下:

第一条　任何组织或者个人不得利用计算机信息系统从事危害国家利益、集体利益和公民合法利益的活动,不得危害计算机信息系统的安全。

计算机信息系统的安全保护应当保障计算机及其相关的和配套的设备、设施(含网络)的安全,保障运行环境的安全,保障信息的安全,保障计算机功能的正常发挥,以维护计算机信息系统的安全运行。

第二条　各计算机信息系统使用单位要建立防病毒、防非法侵入及破坏数据、防其他事故的安全管理制度,明确责任人。

第三条　计算机信息系统进入国际联网,依法由计算机信息系统的使用单位报公安机关备案,并报计算机安全管理领导小组及保卫处备案。

第四条　接入网络的使用单位必须有相应的技术人员和管理人员,具有健全的安全保密管理制度和技术保护措施,符合法律法规所规定的条件,并负责本单位及其用户有关国际联网的技术培训和管理教育工作。

第五条　计算机网络的每个用户必须遵守以下守则:

① 计算机信息系统的建设和应用,应当遵守法律、行政法规和国家其他有关规定。

② 我院计算机软、硬件资源仅用于正常的教学、科研和其他业务工作范围。通过网络系统进行的数据传输、邮件通信或新闻发布,其内容不得违反有关法律法规和学院的有关规定。

③ 计算机网络的用户对自己享用的资源负有保护责任。口令密码应定期更改,不得泄漏给外人,如泄漏,应及时改变口令密码。

④ 计算机的目录和一般文件应设置于可阅读或可执行级别。

⑤ 严禁私自在计算机上进行大量消耗资源且没有科学意义的测试操作、广播型或键式通信操作、游戏或赌博型操作。

⑥ 禁止将外来的非法软件拷入网络,应对重要文件属性加以控制,对远程工作站的登录权限严格控制。国家秘密信息不得在与国际网络联网的计算机信息系统中存储、处理、传递。

⑦ 禁止在网上故意传播病毒,擅自进入他人系统;禁止研究、破译他人的计算机口令,不得私自阅读他人通信文件;防止将带病毒的文件输入计算机。

⑧ 利用国际联网获得资源,特别是软件,应当遵守有关知识产权的法律法规。

⑨ 任何跨国界的信息传输,都是涉及进出口的行为,应当遵守进出口双方国家的法律,都必须为我国有关法律规定所允许;不得利用国际联网从事危害国家安全、泄漏国家秘密等违法犯罪活动;禁止利用网络观看、制作、复制、传播淫秽的、种族主义的、违反宗教法规的和破坏社会稳定的等违反社会公德及国家法律法规的信息。

⑩ 禁止对数据的侵犯及对在线数据库和非在线数据库中的数据滥用。

第六条　第五条第 1,5,6,7,9,10 项的有关内容也适用于单机系统。

第七条　发现计算机信息系统中有违反国家法律法规的内容或发现使用者有违反国家法律法规的行为应及时向院保卫处报告;计算机信息系统中发现有病毒、偷改数据等应及时向院计算机安全管理领导小组报告。

第八条　各部门应做好计算机信息系统的安全保护工作,并协助有关部门查处危害计算机信息系统安全和利用计算机信息系统危害国家利益、集体利益和他人合法利益的违法犯罪行为。

3) 第三、四层是物理实体的安全与硬件系统保护

计算机物理上安全与硬件系统的保护是信息系统安全不可缺少的重要环节。一是必须对自然灾害加强防护:防火、防水、防雷击;二是采取必要的措施防止计算机设备被盗,例如:固定件、添加锁、设置警铃、刻上标签、购置机柜等;三是尽量减少对硬件的损害,例如:消除静电、系统接地、键盘安全套、杜绝电磁干扰信号;另外最好准备不间断电源 UPS (uninterruptible power supply),因为所有电源都有中断的时候,但是如果用了 UPS 电源,在停电时 UPS 能立即切换到内部电池供电,防止断电时计算机立即断电,同时提供一个临时电源,以使你坚持到供电恢复。

4) 第五～七层是网络、软件、信息安全

通信网络、系统软件、信息安全保密技术,是计算机信息系统安全的关键,也是信息安全技术主要的研究内容。本书第 5 章、第 6 章将对此进行详细论述。

3. 信息安全保密研究内容介绍

计算机信息保密与系统安全的研究内容包括:数据加密解密算法、密码分析、密钥管理、操作系统安全、数据库安全、通信网络安全、病毒防治等,下面分别作一简单介绍。

1) 数据加密解密算法

通常,我们将源信息称之为明文,为了保护明文,可以将其通过某种方式变换成无法识别的密文,这个变换处理的过程称之为加密;另一方面,密文可以经过相应的逆变换再还原成为明文,这个变换处理过程称之为解密,如图 1.3 所示。

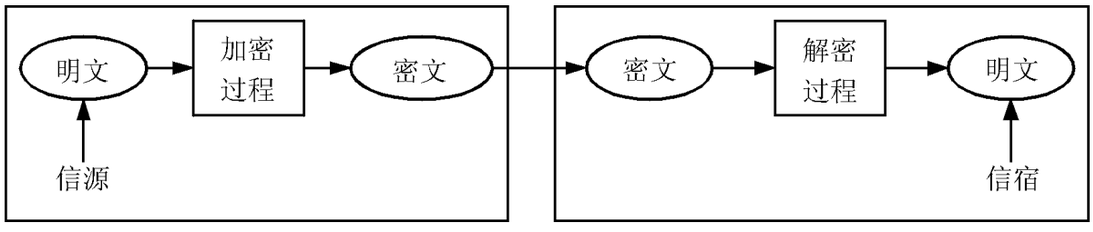


图 1.3　加密解密通信模型

保密算法是用于加密解密变换的数学函数,通常使用两个相关的函数,解密函数是加密函数的反函数,反之亦然。

假设 m(message)代表明文, c(ciphertext)代表密文, E(enciphering)代表加密变换, D(deciphering)代表解密变换,则上述加密解密过程可以用数学形式描述为

$$c = E(m) \qquad (1.1)$$

$$m = D(c) \qquad (1.2)$$

经过加密变换后的信息,即使被偷窃,窃取者由于没有解密手段而无法理解其含义,这样起到了保护源信息的作用。而信息的合法接收具有解密手段,从而可以获得明文信息。当然,为了使密文信息安全可靠,要经常更新算法,增加算法的安全强度,这就是密码学研究的中心内容。

2) 密钥管理

密钥(key)是由数字、字母或特殊符号组成的字符串,它可以控制加密解密的过程。密码算法的安全强度,在很大程度上信赖于密钥的安全保护,这就是密钥管理的研究内容。密钥管理是一门综合性技术,它涉及到密钥的产生、分配、存储、修改以及销毁的全过程,同时还与密钥的行政管理制度与人员的素质密切相关。使用密钥后,加密解密通信模型如图 1.4 所示。

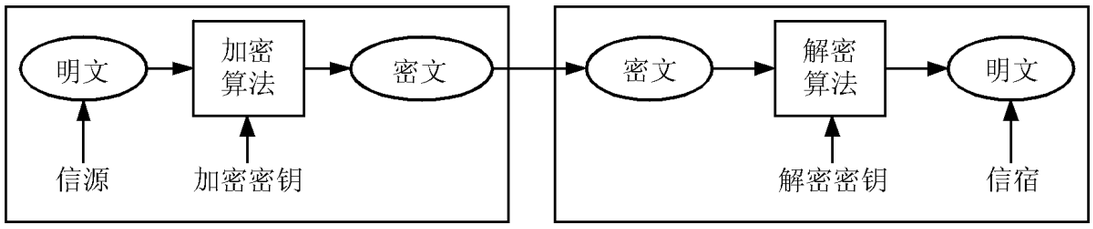


图 1.4　使用密钥后的加密解密通信模型

假设加密密钥为 K_1,解密密钥为 K_2,则数学表达式(1.1)和(1.2)可描述为

$$c = E_{K_1}(m)$$

$$m = D_{K_2}(c)$$

且具有 $D_{K_2}(E_{K_1}(m)) = m$ 的性质。

这样,一个完整的密码系统将由算法、密文、密钥组成,而且密码系统的安全性取决于密钥的保护,算法可以公开,只要严格保管好密钥,破译者就无法将密文解密。

3) 密码分析

密码分析又称为密码破译,它研究在不知道密钥的情况下,通过获取密文而恢复明文的科学。这显然是信息对抗的重要形式,成功的密码分析可能会直接破译明文和密钥。密码分析包括下列类型:

· 仅知密文　攻击者只掌握密文,通过求解方程组得到明文。

已知:

$$c_1 = E_K(m_1)$$

$$c_2 = E_K(m_2)$$

$$\vdots$$

$$c_i = E_K(m_i)$$

求: 明文 $m_1, m_2, \ldots, m_i$ 或密钥 K

· 已知明文　攻击者不仅得到密文,而且也得到了部分密文对应的明文,目标是推导

出密钥或利用该密钥的加解密的算法。

· 选择明文　攻击者不仅得到一些密文和对应的明文,而且能够选择特殊的明文,这样用经过选择后的明文加密,可以得到更多与密钥相关的信息,从而推导出密钥或解密算法。

已知: m_1, $c_1 = E_K(m_1)$

m_2, $c_2 = E_K(m_2)$

·

·

·

m_i, $c_i = E_K(m_i)$　　$m_1, m_2, \ldots, m_i$ 由攻击者选择

求: 密钥或能由 $c_{i+1} = E_K(m_{i+1})$推导出 m_{i+1}的算法。

除此之外,密码分析还有选择密文的攻击、选择密钥的攻击、以及直接从知情者手中获取等多种方式。随着加密技术的发展,密码破译技术也会不停的发展。

4) 计算机系统安全

使用符合安全标准的计算机系统,是维护信息系统安全的技术保障。计算机系统安全包括:设备安全、信息安全和服务安全。设备安全主要是避免对系统硬软实体的破坏和窃取;信息安全重点是保护系统内信息机密、真实和完整;服务安全主要任务是防止硬软设备故障,避免系统服务和功能的丧失。计算机系统安全层次结构如图 1 .5 所示。

计算机系统安全包括如下几个方面。

图 1.5　计算机系统安全层次模型

图 1.6　操作系统的层次结构

(1) 操作系统安全

操作系统是计算机重要的系统软件,它控制和管理计算机系统所有的硬软资源,是计算机系统的指挥中枢。由于操作系统的重要地位,使攻击者常常以操作系统为主要攻击目标,因此研究保护操作系统的方法、设计安全的操作系统,对整个计算机系统的安全至关重要。图 1.6 给出了安全操作系统的层次结构。

(2) 数据库安全

数据库是相关信息的集合。计算机系统中的信息,通常以数据库的形式组织存放,攻

击者通过非法访问数据库,达到篡改和破坏信息的目的。因此,研究如何使数据库记录保密、完整、可用,确保数据库系统安全,成为整个计算机信息系统安全的重要组成部分。图 1.7 给出了安全数据库系统的层次结构。

图 1.7　安全数据库系统的层次结构

(3) 网络安全

计算机网络就是将分散在不同地理位置的计算机系统,通过某种介质连接起来,实现信息和资源的共享。目前,网络化已成为计算机发展的必然趋势。国际互联网 Internet 的普及,带动了世界各地计算机通信网络的发展,同时也为网络系统的安全保密提出了更高的要求。国际标准化组织 ISO(international standand organization),于 1988 年为开放系统互联 OSI(open system interconnetion)增加了安全性要求,提出了八种安全机制,以满足网络安全服务的要求,表 1 .1 给出了安全机制与安全服务的对应关系。

表 1.1　网络安全机制与安全服务的关系

序号	安全机制 / 安全服务	网络加密	数字签名	存取控制	数据完整	鉴别交换	信息填塞	路由控制	公证仲裁
1	对等实体鉴别	Y	Y	N	N	N	N	N	N
2	存取控制服务	N	N	Y	N	N	N	N	N
3	连接机密性	Y	N	N	N	N	N	Y	N
4	无连接机密性	Y	N	N	N	N	N	Y	N
5	选择字段机密性	Y	N	N	N	N	N	N	N
6	信息流安全	Y	N	N	N	N	Y	Y	N
7	可恢复连接完整性	Y	N	N	Y	N	N	N	N
8	无恢复连接完整性	Y	N	N	Y	N	N	N	N
9	选择字段连接完整性	Y	N	N	Y	N	N	N	N
10	无连接完整性	Y	Y	N	Y	N	N	N	N
11	选择字段无连接完整性	Y	Y	N	Y	N	N	N	N
12	数据源鉴别	Y	Y	N	Y	N	N	N	Y
13	源点非否认	N	Y	N	Y	N	N	N	Y
14	交付非否认	N	Y	N	Y	N	N	N	Y

应该强调的是,近年来防火墙(firewall)技术已成为网络安全的重要手段,被网络用户普遍应用,详细内容见第 6 章。

(4) 病毒防治

计算机病毒是一种危害极大的非法程序,它直接威胁着计算机系统的安全。计算机病毒在一定条件下被激活后,立刻感染计算机系统的引导区,或数据文件与程序,侵占系

统资源、毁坏系统信息、降低工作效率,严重者造成整个计算机系统瘫痪。因此研究计算机病毒产生的危害,及时检测和清除病毒,是计算机系统安全不可缺少的组成部分。

1.3 计算机信息系统安全法规和机构

为了维护计算机信息系统的安全,单纯凭技术力量解决是不够的,还必须依靠政府和立法机构,制定出完善的法律和法规进行制约。只有全社会行动起来共同努力,才能从根本上治理高科技领域的犯罪行为,确保计算机信息系统的应用与发展。本节将介绍有关计算机信息系统安全的法规与著名的机构。

1. 计算机信息系统安全的法规

信息系统安全是个国际化的问题,每年全球因计算机损失高达几百亿美元,预计到2000年会上升到2000亿美元。因此,各国政府都非常重视对计算机违法犯罪行为的制裁,制定了许多关于计算机信息系统安全方面的法律规范。我国政府从20世纪80年代初,开始注重这方面的研制工作,1983年在公安部成立了计算机管理监察局,专门负责全国的计算机安全工作,并且陆续颁布了与计算机信息系统安全有关的行政法规,主要有:

- 《电子计算机系统安全规范》(试行草案) 1987年10月
- 《计算机软件保护条例》 1991年5月
- 《计算机软件著作权登记办法》 1992年4月
- 《中华人民共和国计算机信息系统安全保护条例》 1994年2月,见附录一。
- 《计算机信息系统保密管理暂行规定》 1998年2月,国家保密局,见附录二。

这里,1994年2月颁布的《中华人民共和国计算机信息系统保护条例》(以下简称《条例》)是我国第一部计算机安全法规,标志着我国计算机信息系统安全工作已经走上了法制化的轨道,它将成为我国今后计算机安全工作的总纲领。

《条例》的第一部分总则,强调了制定《条例》的对象、目的、范围,以及上级各主管部门的职责,明确表达了计算机信息系统的安全保护应当保障计算机及其相关的设备、设施(含网络)的安全及运行环境的安全,保障信息的安全,保障计算机功能正常发挥,重点维护国家事务、经济建设、国防建设、尖端科学技术等重要领域的信息系统安全。

《条例》的第二部分安全保护制度,指出计算机信息系统建设与应用必须遵守法律,如行政法规,发生有关的案件后,要及时报告上级公安部门,另外邮寄、携带、运输计算机信息媒体,以及销售安全产品的管理制度。公安部门负责计算机病毒和其他危害社会公共安全信息的防治工作。

《条例》的第三部分安全监督,规定了公安机关对计算机信息系统保护工作应行使的监督权限,发现隐患时,应当及时指导使用单位采取措施,在紧急情况下,可以就涉及计算机信息系统安全的特定事项发布专项通令。

《条例》的第四部分法律责任,阐明了违反本条例的诸类行为,公安机关将处以警告或者停机整顿,对故意输入病毒以及其他有害数据的单位和个人,由公安机关分别处以15 000元和5 000元以下的罚款,构成犯罪的,依法追究刑事责任,参照《中华人民共和国

海关法》、《中华人民共和国治安管理处罚条例》、《中华人民共和国国家秘密法》等法规。

《条例》的最后部分附则,进一步明确了计算机病毒、计算机信息系统安全产品的准确定义,指出军队的计算机信息系统安全保护工作,要遵照军队的有关法规执行。

《中华人民共和国计算机信息系统安全保护条例》见附录一。

另外,由于计算机系统在我国各地应用发展水平不同,各省市自治区政府,为了搞好本地区计算机系统安全,纷纷在国家颁布法律规范的前后,制定出自己的暂行规定。例如:《黑龙江省计算机信息系统安全管理暂行规定》、《天津市计算机病毒控制条例》等,所有这些法规的实施,为计算机信息系统安全保密工作的发展起到了极大的推动作用。

2. 国内外著名安全机构

1) 美国国家安全局 NSA

美国国家安全局简称 NSA(National Security Agency),是美国政府的官方安全机构,该机构建立于 1952 年,隶属于美国国防部。它的主要任务是监听和破译所有对本国信息安全有价值的外国通信。

NSA 还一直从事密码学的研究,一方面研究密码算法,加强本国通信的安全;另一方面研究密码分析技术,监听他国的通信。NSA 被世界公认为是拥有最多的数学家的机构,也是先进计算机设备的最大买主,因此,NSA 在密码学研究领域总是处于领先地位,许多实际应用的密码系统都分别被其击破。

2) 美国国家计算机安全协会 NCSA

美国国家计算机安全协会简称 NCSA(National Computer Security Association),它是 NSA 的一个分支机构,担负着国家重要的计算机程序设计工作。该协会负责对商业性的安全产品进行评估,包括硬件产品和软件产品的评估,主持重点项目的研究工作,开展技术指导咨询,提供建议方案,并且组织培训服务。

NSCA 成功地制定出国防部计算机系统的评估准则 DDTCSEC(department of defense trusted computer system evaluation criteria),准则中将安全的可靠性分成 A,B,C,D 四类 8 个等级,具体如下:

D: 最低安全要求,属非安全保护类,它不能用于多用户环境下的敏感信息的处理。只有一个级别。

C: 自主型保护类,它分为两级

C1: 具有一定的自主型存取控制机制,通过用户与数据隔离措施满足安全要求。

C2: 可控制的安全保护机制,通过注册、审查、资源隔离达到安全要求。

B: 强制型安全保护类,它分为三级

B1: 标记安全保护,具有 C2 级的全部功能,并增加了标记强制型访问控制等功能。

B2: 具有形式化安全模型,系统设计结构化,并要求计算机系统加入一种允许用户去评价系统满足哪一级的方法。

B3: 安全区域级,具有严格的系统结构化设计,并具备全面的存取控制的访问监控机制,以及审计报告机制。

A: 验证型安全保护类,分两级

A1: 验证设计,要求用形式化设计说明和验证方法对系统进行分析。

超 A1: 验证客观级,比 A1 级具有更高的安全可信度要求,其技术有待于今后进一步研究探讨。

通常,不同的安全等级,应采取不同的安全技术措施实现,表 1.2 是计算机安全等级的技术策略。

表 1.2 计算机安全等级的技术策略

编号	安全技术策略	A1	B3	B2	B1	C2	C1	D
1	鉴别使用口令登录的各用户					○		
2	维护用户资格不受侵害				○			
3	能产生保持客体存取的审核踪迹					○		
4	受权读取审核踪迹					○		
5	具有事件审核记录					○		
6	用户标识符选择					○		
7	安全状况审核				○			
8	隐蔽信道事件审核			○				
9	实时威胁监控		○					
10	用户存取控制					○		
11	存取未准锁定					○		
12	存取授权限制					○		
13	存取控制表的存取方式和控制		○					
14	资源存储区保护隔离					○		
15	地址空间进程隔离				○			
16	硬件积木化			○				
17	存储区清除再使用							
18	正确标识安全等级					○		
19	打印标志					○		
20	级别信道隔离					○		
21	级别信道路由指定					○		
22	多信道报文标签					○		
23	敏感标志					○		
24	外部资源标志				○			
25	动态终端标识				○			
26	安全、泄漏和完整性					○		
27	对外部用户控制				○			
28	操作、管理人员分离				○			
29	管理活动审核		○					
30	可注册途径				○			
31	安全级别变化可信途径				○			
32	安全恢复		○					

3) 美国国家标准技术学会 NIST

美国国家标准技术学会简称 NIST(National Institute of Standard and Technology),是隶属于美国商业部的一个安全机构。NIST 拥有先进的计算机系统实验室,主要从事开放标准和互操作方面的研究,并对以计算机为基础的工业发展起到了促进作用。NIST 颁布了美国计算机系统采纳的标准与指南,公布的标准是联邦信息处理标准 FIPS(federal information processing standard)。

美国政府在 1988 年颁布的计算机安全条例中,授权 NIST 制定一些在政府机关的计算机系统中,敏感信息的安全标准,并允许 NIST 与其他相关机构一起共同工作。

NIST 地址: Technology Building B-64
Guithersburg, MD 20899
U .S .A

4) RSA 数据安全有限公司 RSADSI

RSA 是一种有名的公开钥密码体制,RSA 数据安全有限公司 RSADSI(RSA Data Security Inc)成立于 1982 年,主要负责开发研制、审批销售 RSA 专利,它也有一些商业产品,包括独立的电子邮件、软件包和各种密码库。

5) 国际密码学研究协会 IACR

国际密码学研究协会简称为 IACR(International Association for Cryptography Research),是世界范围的密码学研究机构,该机构以促进密码学及相关领域理论与实践发展为宗旨,努力推动全球密码界的联系与发展。另一个是 Eurocrypt,每年 5 月在欧洲举行,即著名的欧密会。再有,协会还创办了两个季刊,分别名为“The Journal of Crytology”和“IACR Newsletter”。

6) 信息处理系统安全保护技术委员会

信息处理系统安全保护技术委员会(Technical Committe of Security and Protection in Information Processing System)简称计算机安全技术委员会 TC11,它是国际信息处理联合的第 11 技术委员会,委员会按照主题,分成八个工作组,每组研究的范畴与目的任务各不相同,见图 1.8。

图 1.8 TCⅡ委员会八个工作组

除上述介绍的安全机构以外,还有许多直接与计算机信息系统安全相关的机构,下面

仅列出名称与地址:

· 国际信息系统安全认证集团公司 ISC2(International Information System Security Certification Consortium ,Inc)

地址: P .O .BOX 98

Spencer ,MA 01562 - 0098

· IEEE(IEEE Social Impact Gourp ,Computer Security)社会影响组,计算机安全

地址: 1730 Massachsetts Ave .Washington,DC 20036 - 1903

· 美国计算机安全协会 CSI(Computer Security Institute)

地址: 600 Harrison Street

San Francisco,CA 94107

· 美国信息系统审计与控制协会 ISACA(Information System Audit and Control Association)

地址: P .O .BOX 88180,300 Schmale

Carol Stream ,IL 60188 - 0180

· 美国信息系统安全协会 ISSA(Information System Security Association)

地址: 401 North Michigan Avrbue

Chieago,IL 60611

· 美国国家计算机犯罪数据中心 NCCCD(National Centre Computer Crime Data)

地址: 904 Dabiel Court

Sunta Cruz ,CA 95062

U .S .A

Internet 提供了成千上万的安全参考资料,下面给出一些网址:

www .ibm .com/ IBM 公司的主页面

www .microsoft .com/ Microsoft 公司的主页面

www .coast .net/ simtel/ win3/ security .html 安全档案库

www .jumbo .com/ util/ win/ virus 防病毒软件

www .apple .com/ Apple 公司主页面

misbss20 .larc .nasa .gov/ security/ 4 .0/ macdef .btml 防病毒软件

kaos .deepcove .com/ pacifibyte/ - Virus/ Virusaps .html Mac 防病毒工具

www .stanford .edu/ hom/ computing .html 斯坦福大学的大量资料

www .rsa .com RSA 加密

www .jnt .ac .uk/ newsfiles/ janinfo/ cert/ JANET-CERT/ SOFTWARE .html

UNIX 及网络安全工具

www .awpi .com/ IntelWeb/ canada/ CSIS/ index .html 加拿大安全情报局(CSIS)

第 2 章　预 备 知 识

密码学是通信技术、计算机科学和应用数学之间的边缘科学,数学和计算机科学是其重要的工具,涉及到若干理论分支。数论、信息论和算法复杂性理论更是近代密码学的理论基础。因此,本章对其作简要的介绍。

2.1　数 论 基 础

数论是一门古老的数学分支。以前人们都认为它是完全纯粹的数学,在现实生活中很难找到它的实际应用。自从 1976 年公开密钥密码体制诞生以来,现代密码学就和数论有着千丝万缕的联系。因此,我们先简单介绍一下与密码学有关的数论基本概念。

1. 引言

我们约定:字母 N 表示全体自然数集合, Z 表示全体整数集合,即

$$N = \{0, 1, 2, \cdots\}$$

$$Z = \{\cdots, -2, -1, 0, 1, 2, \cdots\}$$

定义 2.1.1　如果存在一个整数 $k \in Z$ 使得 $n = kd$,则称 d 整除 n,记作 $d \mid n$,其中 d 称作 n 的因数, n 称作 d 的倍数。如果不存在这样一个整数 $k \in Z$ 使得 $n = kd$,,则称 d 不整除 n,记作 $d \nmid n$。

定义 2.1.2　整数 $p(>1)$,称为素数,如果除了 1 和其本身外, p 没有任何其他因数。不是素数的整数称为合数。

例 2.1　$6 = 2 \times 3$, 6 是合数, $2 \mid 6$, 2 是 6 的因数, 6 是 2 的倍数。$7 = 1 \times 7$,除了 1 和 7 之外,没有其它因数,因此 7 是素数。

定理 2.1.1　(带余数除法)设 a, b 是两个整数,其中 $b > 0$。则存在两个整数 q, r 使得

$$a = bq + r \qquad 0 \leq r < b$$

成立,其中 q 和 r 是唯一确定的。

设 a, b 是两个整数。既是 a 的因数又是 b 的因数的数称为 a, b 的公因数, a 和 b 的所有公因数中最大者,称为 a 和 b 的最大公因数,记作 $\gcd(a, b)$。既是 a 的倍数又是 b 的倍数的数称为 a 和 b 的公倍数, a 和 b 的所有公倍数中的最小者称为 a 和 b 的最小公倍数,记作 $\mathrm{lcm}(a, b)$。显然 a 和 b 的最大公因数与最小公倍数满足下列等式:

$$\mathrm{lcm}(a, b) \cdot \gcd(a, b) = ab$$

如果对两个整数 a, b 有 $\gcd(a, b) = 1$,则称 a 与 b 互素。

定理 2.1.2　设 $a, b \in N$,则存在两个整数 u 和 v 使得

$$ua + vb = \gcd(a, b)$$

定理 2.1.3(算术基本定理) 任何一个正整数 m 都存在唯一的因数分解形式

$$m = p_1^{e_1} p_2^{e_2} \cdots p_n^{e_n}$$

其中 $e_i \in N$, p_i 是素数且 $p_1 < p_2 < \cdots < p_n$。

这个分解形式也称 m 的标准分解形式。

例 2.2 $6 = 2 \times 3, 20 = 2^2 \times 5, 100 = 2^2 \times 5^2$

有了算术基本定理后,就可以把任何整数分解成标准形式,从而可以方便地求出两个整数的最大公因数和最小公倍数。设 a,b 是两个正整数,且有标准分解形式:

$$a = p_1^{e_1} p_2^{e_2} \cdots p_n^{e_n}, \quad b = p_1^{f_1} p_2^{f_2} \cdots p_n^{f_n}, \quad e_i, f_i \in N$$

则

$$\gcd(a, b) = \prod_{i=1}^{n} p_i^{\min\{e_i, f_i\}}$$

$$\mathrm{lcm}(a, b) = \prod_{i=1}^{n} p_i^{\max\{e_i, f_i\}}$$

其中 $min\{x,y\}$表示 x,y 中最小者, $max\{x,y\}$表示 x,y 中最大者。

2. Euclid 算法

利用算术基本定理,原则上可以求得任何两个整数的最大公因数,但当两个整数比较大时求它们的标准分解形式非常困难,目前还没有有效的算法,因此求它们的最大公因数也变得非常困难。Euclid 算法从另一方面解决了求两整数的最大公因数的问题。

Euclid 算法又称辗转相除算法,由带余数除法,有下列等式:

$$\begin{aligned}
a &= bq_1 + r_1 && 0 < r_1 < b \\
b &= r_1 q_2 + r_2 && 0 < r_2 < r_1 \\
&\cdots \\
r_{n-2} &= r_{n-1} q_n + r_n && 0 < r_n < r_{n-1} \\
r_{n-1} &= r_n q_{n+1} + r_{n+1} && r_{n+1} = 0
\end{aligned}$$

因为每进行一次带余数的除法,余数至少减 1,而 b 是有限的。所以,最多进行 b 次带余数的除法,总可以得到一个余数是 0 的等式,即最后一个等式,而最后一个不等于 0 的余数 r_n 就是我们要求的 a 和 b 的最大公因数 $\gcd(a, b)$。

从上面的 Euclid 算法中可以看出,将 $r_1 = a - bq_1$ 代入第二个等式中,将 $r_2 = b - r_1 q_2$ 代入到第三个等式中,…,将 $r_{n-1} = r_{n-3} - r_{n-2} q_{n-1}$代入倒数第二个等式中,就可以得到 r_n 关于 a, b 的一个表示式,其中 a, b 的系数分别就是定理 2.1.2 中的 u, v。

例 2.3 求 $\gcd\{1694, 917\}$

$$\begin{aligned}
1694 &= 1 \times 917 + 777 \\
917 &= 1 \times 777 + 140 \\
777 &= 5 \times 140 + 77 \\
140 &= 1 \times 77 + 63 \\
77 &= 1 \times 63 + 14
\end{aligned}$$

$$63 = 4 \times 14 + 7$$

$$14 = 2 \times 7 + 0$$

所以 $\gcd\{1694, 917\} = 7$

进行回代

$$\begin{aligned}
7 &= 63 - 4 \times 14 \\
&= 63 - 4 \times (77 - 63) \\
&= -4 \times 77 + 5 \times 63 \\
&= -4 \times 77 + 5 \times (140 - 77) \\
&= 5 \times 140 - 9 \times 77 \\
&= 5 \times 140 - 9 \times (777 - 5 \times 140) \\
&= -9 \times 777 + 50 \times 140 \\
&= -9 \times 777 + 50 \times (917 - 777) \\
&= 50 \times 917 - 59 \times 777 \\
&= 50 \times 917 - 59 \times (1694 - 917) \\
&= -59 \times 1694 + 109 \times 917
\end{aligned}$$

即 $3 = u \times 1694 + v \times 917$

其中 $u = -59, v = 109$

3. 同余

定义 2.1.3 假设 a 和 b 是两个整数，m 是一个正整数，如果 $m \mid b - a$，则称 a 与 b 对模 m 同余。记作 $a \equiv b \pmod{m}$。

例 2.4 3 和 1 对模 2 同余，4 和 1 对模 3 同余，即 $3 \equiv 1 \pmod 2$，$4 \equiv 1 \pmod 3$

定理 2.1.4 同余的性质

设 a, b, c 和 m 是整数，则有：

1) $a \equiv a \pmod{m}$

2) 若 $a \equiv b \pmod{m}$，则 $b \equiv a \pmod{m}$

3) 若 $a \equiv b \pmod{m}$，$b \equiv c \pmod{m}$，则 $a \equiv c \pmod{m}$

假设 a 和 b 被 m 除，获得整数商和余数，这个余数在 0 和 $m - 1$ 之间，即 $a = q_1 m + r_1$ 和 $b = q_2 m + r_2$，$0 \leq r_1 \leq m - 1$，$0 \leq r_2 \leq m - 1$。不难看出，$a \equiv b \pmod{m}$，当且仅当 $r_1 = r_2$。我们使用符号 $a \pmod{m}$ 来表示 a 被 m 除时的余数，即上面的 r_1，这样 $a \equiv b \pmod{m}$，当且仅当 $a \pmod{m} \equiv b \pmod{m}$。如果我们用 $a \pmod{m}$ 来代替 a，我们说 a 是被模 m 约简的。

现在我们能够定义模 m 的算术：Z_m 是一个集合 $\{0, 1, \cdots, m - 1\}$，它有两种运算 + 和 ×。在 Z_m 中的加法和乘法，除了将结果被模 m 约简外，恰好像实数加法和乘法。

例 2.5 在 Z_2 中作加法

$0 + 0 \equiv 0 \pmod 2$，$0 + 1 \equiv 1 \pmod 2$，$1 + 0 \equiv 1 \pmod 2$，$1 + 1 \equiv 0 \pmod 2$

一般地，在 Z_2 中的加法称为模 2 加，有时也称比特异或，用记号 $\oplus$ 表示。

$$0 \oplus 0 = 0, 0 \oplus 1 = 1, 1 \oplus 0 = 1, 1 \oplus 1 = 0$$

例 2.6　在 Z_{16} 中作乘法 11×13

$$11\times13=143$$
$$=8\times16+15$$

所以
$$11\times13\pmod{16}=15$$

定义 2.1.4　Euler 函数是定义在正整数上的函数，它在正整数 m 的值等于 $1,2,\cdots,m-1$ 中与 m 互素的数的个数，记作 $\phi(m)$。

例 2.7　$m=6$，$\{1,2,3,4,5\}$ 中与 m 互素的数为 $\{1,5\}$，共有两个，所以

$$\phi(m)=\phi(6)=2$$

定理 2.1.5　设正整数 m 的标准分解形式为

$$m=p_1^{e_1}p_2^{e_2}\cdots p_n^{e_n},$$

则
$$\phi(m)=m\left(1-\frac{1}{p_1}\right)\left(1-\frac{1}{p_2}\right)\cdots\left(1-\frac{1}{p_n}\right)$$

例 2.8　$m=6$，其标准分解形式为 $6=2\times3$

所以
$$\phi(6)=6\left(1-\frac{1}{2}\right)\left(1-\frac{1}{3}\right)=2$$

定理 2.1.6（Euler 定理）　若 a 和 m 互素，则

$$a^{\phi(m)}\equiv1\pmod{m}$$

推论　（Fermat 定理）若 p 是素数，则

$$a^p\equiv a\pmod{p}$$

设 $f(x)$ 表示多项式：$a_nx^n+a_{n-1}x^{n-1}+\cdots+a_0$，其中 $a_n\neq0$，$a_i\in N(i=1,2,\cdots,n)$。设 m 是一个正整数，则

$$f(x)\equiv0\pmod{m}$$

称作模 m 的同余式，n 称为同余式的次数，$n=1$ 时称为一次同余式，$n=2$ 时称为二次同余式。

若 a 是使 $f(a)\equiv0\pmod{m}$ 成立的一个整数，则 $x\equiv a\pmod{m}$ 叫作同余式的一个解。

定理 2.1.7（中国剩余定理）　设 $m_1,m_2,\cdots,m_k$ 是 k 个两两互素的整数，$m=m_1m_2\cdots m_k$，$M_i=m/m_i$，$i=1,2,\cdots,k$。则同余方程组

$$x\equiv b_1\pmod{m_1},\quad x\equiv b_2\pmod{m_2},\quad\cdots,\quad x\equiv b_k\pmod{m_k}$$

有解 $x\equiv M_1'M_1b_1+M_2'M_2b_2+\cdots+M_k'M_kb_k\pmod{m_k}$

其中 $M_i'M_i\equiv1\pmod{m_i}$，$i=1,2,\cdots,k$。

由此定理可以看出，M_i' 可以利用前面介绍的 Euclid 算法求出。

4. 二次剩余

设 $\gcd(a,m)=1$，若同余式 $x^2\equiv a\pmod{m}$ 有解，则 a 称作模 m 的二次剩余，否则称作模 m 的二次非剩余。

例 2.9　考虑下列同余式

$$x^2\equiv1\pmod5,\quad x^2\equiv2\pmod5,\quad x^2\equiv3\pmod5,\quad x^2\equiv4\pmod5$$

不难发现：$x=1$，$x=4$ 满足 $x^2\equiv 1(\bmod 5)$

$x=2$，$x=3$ 满足 $x^2\equiv 4(\bmod 5)$

不存在整数 x 满足

$$x^2\equiv 2(\bmod 5),\quad x^2\equiv 3(\bmod 5)$$

所以 1,4 是模 5 的二次剩余，而 2,3 是模 5 的二次非剩余。

定理 2.1.8 若 $\gcd(a,p)=1$，则 a 是模 p 的二次剩余的充要条件为

$$a^{\frac{p-1}{2}}\equiv 1(\bmod p)$$

a 是模 p 的二次非剩余的充要条件为

$$a^{\frac{p-1}{2}}\equiv -1(\bmod p)$$

定理 2.1.9 两个模 p 的二次剩余的乘积或两个模 p 的二次非剩余的乘积，还是模 p 的二次剩余，一个模 p 的二次剩余与另一个模 p 的二次非剩余的乘积是二次非剩余。

定义 2.1.5 Legendre 符号 $\left(\frac{a}{p}\right)$ 是一个对于给定素数 p 定义在一切整数 a 上的函数，它的值规定如下：

$$\left(\frac{a}{p}\right)=\begin{cases}1 & a \text{ 是模 } p \text{ 的二次剩余}\\ -1 & a \text{ 是模 } p \text{ 的二次非剩余}\\ 0 & p\mid a\end{cases}$$

例 2.10 $\left(\frac{1}{5}\right)=\left(\frac{4}{5}\right)=1$，$\left(\frac{2}{5}\right)=\left(\frac{3}{5}\right)=-1$，$\left(\frac{5}{5}\right)=0$

定理 2.1.10 Legendre 符号的性质

1) $\left(\frac{1}{p}\right)=\left(\frac{q^2}{p}\right)=1$

2) 如果 $a\equiv b(\bmod p)$，则 $\left(\frac{a}{p}\right)=\left(\frac{b}{p}\right)$

3) $\left(\frac{a}{p}\right)\equiv a^{(p-1)/2}(\bmod p)$

4) $\left(\frac{ab}{p}\right)=\left(\frac{a}{p}\right)\left(\frac{b}{p}\right)$

5) 设 p,q 均为奇素数，$p\neq q$，则 $\left(\frac{q}{p}\right)\left(\frac{p}{q}\right)=(-1)^{(p-1)(q-1)/4}$

定义 2.1.6 设 $m=\prod_{i=1}^{n}p_i^{e_i}$，$e_i\geq 0$ 是一个奇正整数，u 是与 m 互素的整数，则 Jacobi 符号定义为

$$(u/m)=\prod_{i=1}^{n}\left(\frac{u}{p_i}\right)^{e_i}$$

其中 $\left(\frac{u}{p_i}\right)$ 是 Legendre 符号。

定理 2.1.11 Jacobi 符号的运算性质

1) $(u/m)=(u-m)/m$

2) $(uv/m)=(u/m)(v/m)$

3) $(u/mn)=(u/m)(u/n)$

4) $(-1/m)=1$ 当且仅当 $m\equiv 1(\bmod 4)$

5) $(2/m)=1$ 当且仅当 $m\equiv \pm 1(\bmod 8)$

6) 设 m, n 都为奇数,且 $(m, n)=1$,则 $\left(\frac{n}{m}\right)\left(\frac{m}{n}\right)=(-1)^{(m-1)(n-1)/4}$

2.2 信息论基础

Shannon 信息论是密码学的理论基础,本节介绍 Shannon 信息论的基本概念,与密码学理论有关的概念将在第 3 章介绍。

1. 熵的概念

熵是信息的数学测度或不确定性,它是作为概率分布的一个函数来进行计算的。假设有一个随机变量 x,它根据概率分布 $P(x)$在一个有限集合上取值。根据分布 $P(x)$发生的事件来获得的信息是什么?等价地,如果一个事件还没有发生,有关这个结果的不确定性是什么?这个量称为 x 的熵并用 $H(x)$表示。

定义 2.2.1 离散随机变量 x 的熵 $H(x)$定义为

$$H(x)=-\sum_{x\in X}P(x)\log_2 P(x)$$

其中,$P(x)$表示随机变量 x 的概率分布。

例 2.11 设离散随机变量 x 取$\{0,1\}$两个值,其中 $P(x=0)=P(x=1)=\frac{1}{2}$,则

$$\begin{aligned}H(x)&=-P(x=0)\log_2 P(x=0)-P(x=1)\log_2 P(x=1)\\&=-\frac{1}{2}(-1)-\frac{1}{2}(-1)\\&=1\end{aligned}$$

下面我们来定义联合熵与条件熵。

定义 2.2.2 两个离散随机变量(x, y)的联合熵定义为

$$H(xy)=-\sum_{x\in X}\sum_{y\in Y}P(xy)\log_2 P(xy)$$

其中,$P(xy)$表示离散随机变量(x, y)的联合概率分布。

类似地,可以定义 n 个离散随机变量$(x_1, x_2, \cdots, x_n)$的联合熵。

定义 2.2.3 两个离散随机变量(x, y)的条件熵定义为

$$H(x\mid y)=-\sum_{x\in X}\sum_{y\in Y}P(xy)\log_2 P(x\mid y)$$

其中,$P(xy)$表示离散随机变量(x, y)的联合概率分布,$P(x|y)$表示两离散随机变量的条件分布。

利用联合熵与条件熵的定义,容易证明

定理 2.2.1 $H(xy)=H(x)+H(y|x)$

2. 互信息

互信息是一个事件含有另外一个事件的信息的度量;或者是已知另外一事件(称作 B)的情况下,事件(称作 A)不确定性的减少。

定义 2.2.4 两个离散随机变量 x 和 y,它们具有概率分布 $P(x)$ 和 $P(y)$和联合概率密度 $P(xy)$,则互信息定义为

$$I(x;y) = \sum_{x\in X}\sum_{y\in Y} P(xy)\log_2 \frac{P(xy)}{P(x)P(y)}$$

从互信息的定义可以看出,如果随机变量 x 和 y 统计独立,即 y 不含 x 的任何信息,则 $I(x;y)=0$。

互信息具有对称性,这就是

定理 2.2.2 $I(x;y) = H(x) - H(x|y)$

$$= H(y) - H(y|x)$$

$$= I(y;x)$$

2.3 计算复杂性简介

计算复杂性理论是计算机科学理论的一个分支,它提供了一种分析不同密码技术和算法保密强度的方法。它对密码算法和技术进行比较,然后确定它们的安全性。现代密码学的许多理论和技术是建立在某些计算问题的复杂性基础之上的。

1. 算法复杂性

一个算法的计算复杂性用符号“O”来表示,计算复杂性的数量级是这种类型的函数,即当 n 变大时,增长最快的函数(n 是输入尺寸),所有常数和较低阶形式的函数忽略不计。例如,一个所给的算法复杂性是 $5n^2+8n+10$,那么其计算复杂性表示为 $O(n^2)$。

通常,算法按其时间和空间的复杂性进行分类,如果一个算法的复杂性是不依赖于 n: $O(1)$,那么它是“常数级的”。如果它的复杂性随 n 线性增长,那么它是“线性的”。O 随 n 增长的其他一些算法也称之为“二次方的”,“三次方的”,等等。所有这些算法都是“多项式的”,它们的复杂性是 $O(n^t)$,这里 t 是一个常数。有一个多项式的时间复杂性的算法族称之为多项式时间算法。

复杂性是 $O(t^{f(n)})$的算法,被称为是“指数的”,这里 t 是一个常数,$f(n)$是 n 的多项式函数。

2. 问题的分类

复杂性理论按照解决问题的算法对问题进行分类。能够用多项式时间算法解决的问题称之为易处理的;不能在多项式时间内解决的问题称之为难处理的,难处理的问题有时也称为难解的。

定义 2.3.1 P 类问题:在多项式时间内可以解决的问题。

NP 类问题:多项式时间内可以验证的问题。

由于在多项式时间内可以解决的问题,在多项式时间内也完全可以验证其正确性。因此,一般有 $P \subseteq NP$,但现在还不知道是否有“$P = NP$”成立。

在 *NP* 问题中有些特殊的问题能够被证明与此类中的任何问题一样困难,这类问题称之为 *NP* - 完全类问题。有人已经编辑了一份 *NP* - 完全类问题的目录,下面将列出几个例子。

3. 几个例子

1) 整数分解问题

前面介绍了算术基本定理,根据这个定理,任何一个正整数都可以分解成标准形式

$$m = \prod_{i=1}^{n} p_i^{e_i}, \quad p_i \text{ 是素数}, e_i \in N$$

当 m 较小时,这个问题不太困难,例如

$$6 = 2 \times 3, \quad 100 = 2^2 \times 5^2$$

但当 m 较大时,这个问题就变得非常困难了,例如你能立即指出整数 8616460789 的标准分解式吗?特别当 m 达到几百位时,根据现有的算法用最快的计算机也不行。反过来,如果给定一个整数的标准分解式,则可以很快验证这个标准分解式是否是这个整数的标准分解式。

例 2.12 861646079 的标准分解式为

$$861646079 = 89681 \times 96079$$

我们可以立即验证上述等式成立。

从上面的讨论来看,整数分解问题一般属于 *NP* 类问题。

2) 背包问题

背包问题是这样一个问题:已知长度为 k 的圆形背包及长度分别为 $a_1, a_2, \cdots, a_n$ 的 n 个圆形物品。假定这些物品的半径和背包半径相同,要求从 n 个物品中选出若干个正好装满这个背包。

把背包问题抽象成数学模型,称为子集合问题:设有长度为 n 的向量 $\boldsymbol{A} = (a_1, a_2, \cdots, a_n)$,任意给定一个正整数 k,寻找有没有一些 a_i 的和恰好等于 k,即求方程

$$\sum_{i=1}^{n} x_i a_i = k$$

的解向量 $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)$,其中 $x_i = 0$ 或 1。

当背包向量 $\boldsymbol{A}$ 的长度 n 比较小时,可以用穷举搜索法求得解向量 $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)$;但当 n 比较大时,比如说 200,那么用穷举法就不可行了。

反过来,如果给定一个向量 $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)$,可以很容易验证它是不是背包问题的解,因此背包问题是 *NP* 类问题,计算机理论科学已经证明,背包问题是 *NP*-完全问题。

例 2.13 长度为 10 的背包向量

$$\boldsymbol{A} = \{43,129,215,473,903,302,561,1165,697,1523\}$$

给定整数 $k = 3231$,要求

$$\sum_{i=1}^{10} a_i x_i = 3231$$

的解向量 $\boldsymbol{x}$,需穷举搜索。反过来,如果给定向量 $\boldsymbol{x} = (0,1,0,1,1,0,1,1,0,0)$,我们可以很容易证明

$$129 + 473 + 903 + 561 + 1165 = 3231$$

因此,给定的向量是这个背包问题的解向量。

3) 离散对数问题

设 x, r, n 是正整数。已知 x, r 和 n,可以很快地求得

$$y \equiv x^r (\mathrm{mod}\ n)$$

反过来,如果已知 y, x 和 n,求 r 使得:

$$y \equiv x^r (\mathrm{mod}\ n)$$

成立,这便是离散对数问题。当 y, x 和 n 都比较小时,可以用穷举搜索求得 r,如果这些数都比较大时,这便非常困难了。

如果给定一个整数 r,那么可以很容易验证它是否为

$$y \equiv x^r (\mathrm{mod}\ n)$$

的解。因此离散对数问题是 NP 类问题。计算机理论科学已经证明,离散对数问题是 NP - 完全问题。

第 3 章　传统密码体制

3.1　密码学的基本概念

密码是一门古老的技术,它已有几千年的历史,自从人类社会有了战争就出现了密码,但 1949 年以前的密码只是一种艺术而不是科学,那时的密码专家常常凭直觉和经验来设计和分析密码,而不是靠严格的理论证明。1949 年,Shannon 发表了题为“保密系统的通信理论”一文,引起了密码学的一场革命。在这篇文章中,他把密码分析与设计建立在严格的理论推导基础之上,从而使得密码真正成为一门科学。

密码学按目的来分,可分为密码编码学和密码分析学。密码分析学的基本任务是研究如何破译加密的消息或者伪造消息;密码编码学的目的是伪装消息,就是对给定的有意义的数据进行可逆的数学变换,将其变为表面上杂乱无章的数据,使得只有合法的接收者才能恢复原来有意义的数据,而其余任何人都不能恢复原来的数据。

变换前有意义的数据称为明文,所有可能的明文组成的集合称为明文空间。变换后表面上杂乱无章的数据称为密文,所有可能的密文组成的集合称为密文空间。对明文数据进行可逆变换的过程称为加密过程,其变换称为加密变换,加密变换由一个参数 k_1 控制,这个参数称为加密密钥。恢复明文的变换过程称为解密过程,其变换称为解密变换,它也由一个参数 k_2 控制,这个参数称为解密密钥。

在传统密码体制中,加密密钥 k_1 与解密密钥 k_2 是相同的(或者从 k_1 很容易推导出 k_2),统称为 k。因此,传统密码体制又称为对称密码体制。在现代公开密钥密码体制中,加密密钥 k_1 与解密密钥 k_2 是不相同的,因此公开密钥密码体制中又称为非对称密码体制。本章简要介绍传统密码体制,第 4 章介绍公开密钥密码体制。

3.2　保密系统的 Shannon 理论

1. 保密系统的 Shannon 模型

图 3.1 是保密系统的 Shannon 模型,n 元数组 $x^n = (x_1, x_2, \cdots, x_n)$ 为明文,n 元数组 $y^n = (y_1, y_2, \cdots, y_n)$ 是在公开信道上传送的密文,也就是说任何人都可得到这些密文,n 元数组 $k^n = (k_1, k_2, \cdots, k)$ 是通过安全信道传送给接收者的密钥,敌方密码分析者无法获得该密钥。加密器的任务就是对明文 m 施以加密变换到密文 c

$$c = E_k(m)$$

解密器的任务就是对所接收到的密文施行解密变换获得明文

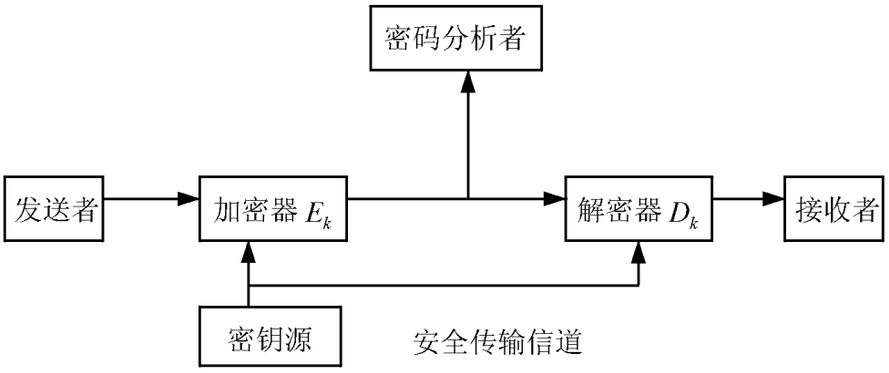


图 3.1　保密系统的 Shannon 模型

$$m' = D_k(c) = D_k(E_k(m))$$

由于加密变换、解密变换是依赖于密钥 k 的一对可逆的数学变换，因此

$$m' = m$$

从而完成保密通信。

例 3.1　一次一密密码体制。

设明文 m 是一串二进制数据：$m = (0110010011)_2$，密钥 k 也是一串同样长度的二进制数据：$k = (0111001001)_2$。在 A，B 两方通信前，A 首先通过安全信道（比如信使）把密钥 k 送给 B，现在 A 要把明文 m 通过公开信道送给 B，A 先对 m 施行加密变换

$$\begin{aligned} c &= E_k(m) = m \quad k \\ &= (0110010011)_2 \quad (0111001001)_2 \\ &= (0001011010)_2 \end{aligned}$$

注：向量的模 2 加　是指每个向量的分量进行模 2 加。

B 收到 c 后，用事先 A 传送给它的密钥 k 进行解密。

$$\begin{aligned} m' &= D_k(c) = c \quad k \\ &= (0001011010)_2 \quad (0111001001)_2 \\ &= (0110010011)_2 \\ &= m \end{aligned}$$

从而 B 获得明文 m。而任何获得密文 c 的密码分析者由于没有密钥 k，因此也就无法获得正确的明文。

2. 理想保密与完善保密

从上面 Shannon 模型可以看出，保密的关键是密钥。任何人只要获得通信密钥，就能够正确地恢复明文。

设明文 $x^n = (x_1, x_2, \cdots, x_n)$，密文为 $y^n = (y_1, y_2, \cdots, y_n)$，密钥为 k，如果对所有的 n 有 $I(x^n; y^n) = 0$，即对所有 n，x^n 与 y^n 统计独立，从 y^n 得不到任何关于 x^n 的信息，这种密码体制称为完善保密的。前面介绍的一次一密的密码体制就是完善保密的。如果

$$0 < I(x^n; y^n) < H(x^n)$$

即 y^n 包含有 x^n 的信息,这时可以推得 $H(k|y^n)>0$,即无论 n 有多大,已知密文 y^n,k 还是不确定的。没有确定的密钥就不能正确地解密,恢复明文,这种密码体制称作理想保密。

3.3 序列密码

密码按加密方式不同可以分为序列密码与分组密码,序列密码是逐个字符加密的,而分组密码是按字符块加密的。

1.序列密码的工作原理

序列密码的工作原理非常直观。假设 $m=m_0\ m_1\ m_2\cdots$ 是一个待加密的明文序列(一般是二进制 0,1 序列),$k=k_0\ k_1\ k_2\cdots$ 是一个与明文序列等长的二元(伪)随机序列,称为密钥序列。收发两端都事先知道密钥序列的内容了,于是在序列密码中,用密钥序列 k 对明文序列 m 进行加密的过程是将 k 和 m 对应的分量进行简单的模 2 相加,得到加密后的明文序列

$$c=c_0\ c_1\ c_2\ \ldots$$

即 $c_i\equiv(k_i+m_i)(\mathrm{mod}\ 2)$。在接收端,合法的接收者的解密过程就是将密文序列 c 和密钥序列的对应分量进行简单的模 2 相加。于是原来的明文序列就恢复出来了,因为

$$m_i\equiv(k_i+c_i)(\mathrm{mod}\ 2)$$

其原理如图 3.2 和图 3.3 所示。

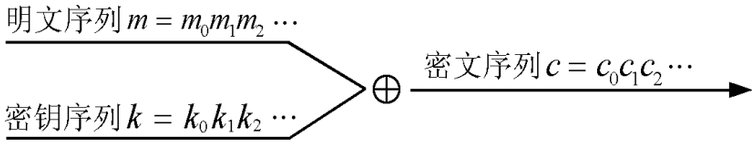


图 3.2 序列密码的加密过程

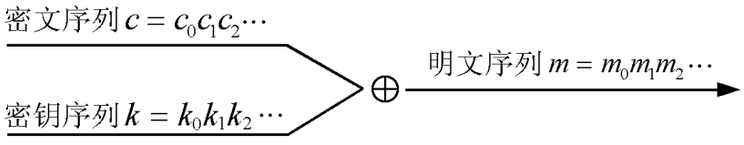


图 3.3 序列密码的解密过程

2.线性移位寄存器(LFSR)

从保密系统的 Shannon 理论和序列密码的工作原理可知,序列密码保密的关键是如何高效地产生安全可靠的二元随机序列作为密钥序列,由于目前还没有有效地产生二元随机序列的实用办法,因此,一般都用伪随机二元序列作为密钥序列。

所谓伪随机序列就是貌似随机序列的序列,或者说很像随机序列的序列。线性移位寄存器就是能够产生这样一种伪随机序列的逻辑电路,它的工作原理如图 3.4 所示。

图 3.4 中,$f(a_{j-n},a_{j-n+1},\cdots,a_{j-1})=c_0\ a_{j-1}\quad c_1\ a_{j-2}\quad\cdots\quad c_{n-1}\ a_{j-n}$ 是线性移位

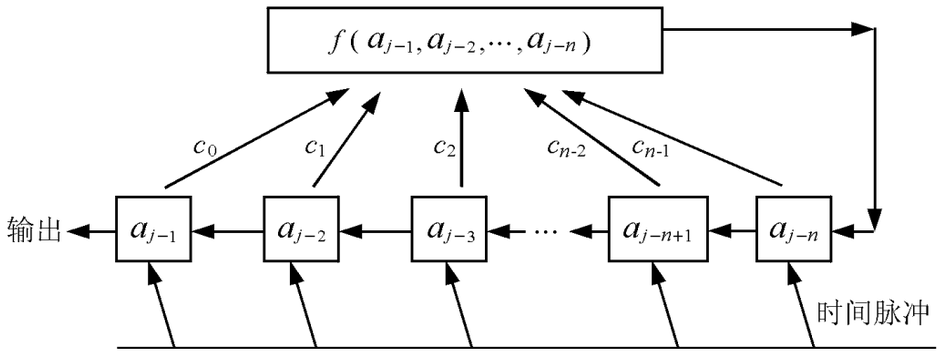


图 3.4　线性移位寄存器

寄存器的反馈函数,或称反馈逻辑。下面的小方框代表寄存器,每个寄存器有两个状态:0或1。n 称为线性移位寄存器的级数。当一个时间脉冲来临时,最左边一个寄存器内的值输出,其余依次往左移动一位,最右边的寄存器则接收反馈函数计算得到的值。当不断地有时间脉冲来临时,最左边输出一串二元伪随机序列 $a_0\ a_1\ a_2\cdots$。

例 3.2　考虑图 3.5 的三级线性反馈移位寄存器。

$$f(a_{j-3},a_{j-2},a_{j-1}) = c_0\ a_{j-3} \quad c_1\ a_{j-2} \quad c_2\ a_{j-1} = a_{j-3} \quad a_{j-1}$$

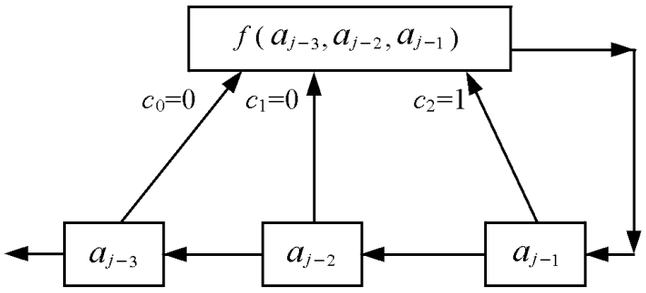


图 3.5　3 级线性移位寄存器

当 $j=3$ 时,取初值 $a_0=0$, $a_1=0$, $a_2=1$;当 $j=4$ 时,一个时间脉冲来临, $a_0=0$ 输出, $a_1=0$ 移到最左边一个寄存器, $a_2=1$ 移到第二个寄存器, $a_3=f(a_0,a_1,a_2)=a_0 \quad a_2=1$ 送到第三个寄存器;当 $j=5,6,\cdots$,即不断地有时间脉冲来临时,最左边就输出一串二元伪随机序列:0 0 1 1 1 0 1 0 0 …。

3. 序列密码的设计

虽然线性移位寄存器能够快速产生伪随机特性比较好的二元随机序列,但在实用中它还不能直接用作序列密码密钥序列,这主要是因为它的高度可以预测。因此,必须对线性移位寄存器的输出序列进行适当的处理,这就是序列密码的设计问题。

1) 非线性组合序列密码

非线性组合序列密码的工作原理如图 3.6 所示。

图 3.6 中左边的 n 个小方框代表 n 个线性反馈移位寄存器,分别输出序列 $\{a_{1i}\},\cdots,\{a_{ni}\}$。右边大方框代表一个 n 元变量的非线性组合函数 $f(x_1,x_2,\cdots,x_n)$,它以 n 个线性反馈移位寄存器的输出序列作为变量

$$b_i = f(a_{1i},a_{2i},\cdots,a_{ni}), \quad i=1,2,\cdots$$

这时序列 $\{b_i\}$ 就可以作为密钥序列了。

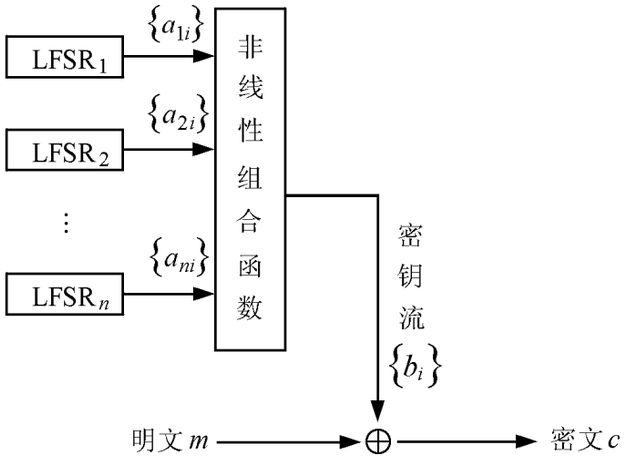


图 3.6　非线性组合序列密码

2) 前馈序列密码和钟控序列密码

除了前面介绍的非线性组合序列密码外,还有前馈序列密码和钟控序列密码,它们的工作原理同非线性组合序列密码比较类似,这里仅列出其工作原理框图,如图 3.7 和图 3.8 所示。欲了解详细情况请参阅有关参考书。

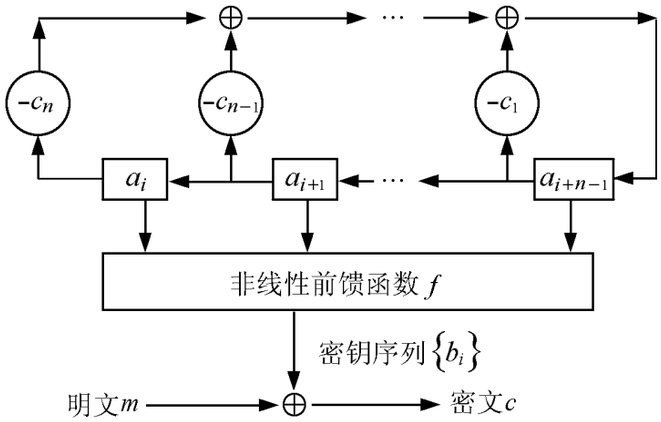


图 3.7　前馈序列密码

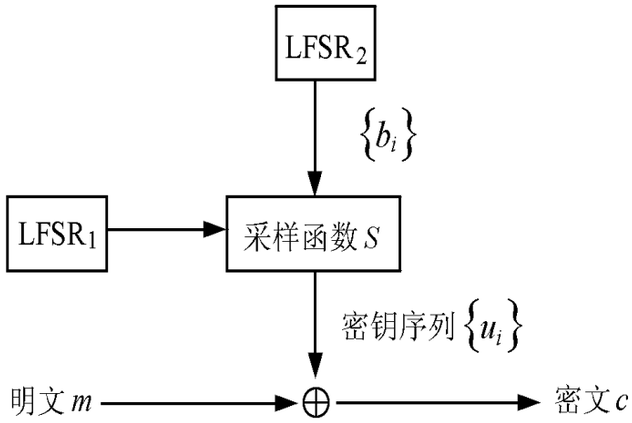


图 3.8　钟控序列密码

3.4　分 组 密 码

1. 分组密码的工作原理

分组密码的工作原理不像序列密码那样直观,它首先将明文分成相同长度的比特块,然后分别对每个比特块加密产生一串密文块。解密时,对每个密文块进行解密得到相应的明文比特块,将所有的明文比特块合并起来即得到明文。如图 3.9 所示。

2. 数据加密标准

数据加密标准,DES(Data Encryption Standard),它是分组密码的一个典型代表。1974 年,美国国家标准局(NBS)向全社会公开征集一种用于政府部门非机密数据的加密算法。IBM 公司提出了一种称为 LUCIFER 的算法,在此基础上,经过一段时间的修改与简化,美国国家标准局于 1977 年正式颁布这个算法,称为数据加密算法 DES,用作政府及商业部门的非机密数据加密标准。

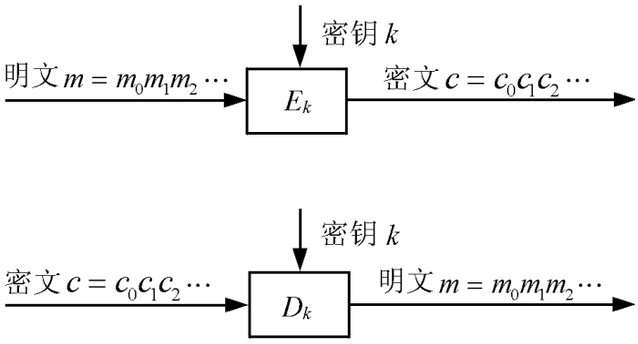


图 3.9　分组密码的工作原理

1) DES 的结构图

DES 用 56 位密钥加密 64 位明文，输出 64 位密文，它的加密过程如图 3.10 所示。

设输入 64 比特明文，先对输入的 64 比特明文进入初始置换 IP(见表 3.1)，置换后明文 58 比特变为第一位，第 50 比特变为第二位，等等，将新得到的 64 比特左边的 32 位记为 L_0，右边的 32 比特记为 R_0。经 16 圈处理以后，再经逆初始置换 IP^{-1}(见表 3.2)产生最后的 64 比特输出密文。

表 3.1　初始置换 IP

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

表 3.2　逆初始置换 IP^{-1}

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

加密过程可用数学公式表示如下：

$$L_i = R_{i-1}$$

$$R_i = L_{i-1} \quad f(R_{i-1}, K_i)$$

$$i = 1,2,3,\cdots,16$$

下面对加密过程中的函数 f 作详细描述。

首先将 R_{i-1}的 32 比特膨胀为 48 比特的向量，膨胀方法见比特选择表 E(见表 3.3)，然后将这 48 比特向量与 48 比特向量 K_i(子密钥)进行模 2 加，得出一个 48 比特向量，将

表 3.3　比特选择表 E

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

表 3.4　置换 P

16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

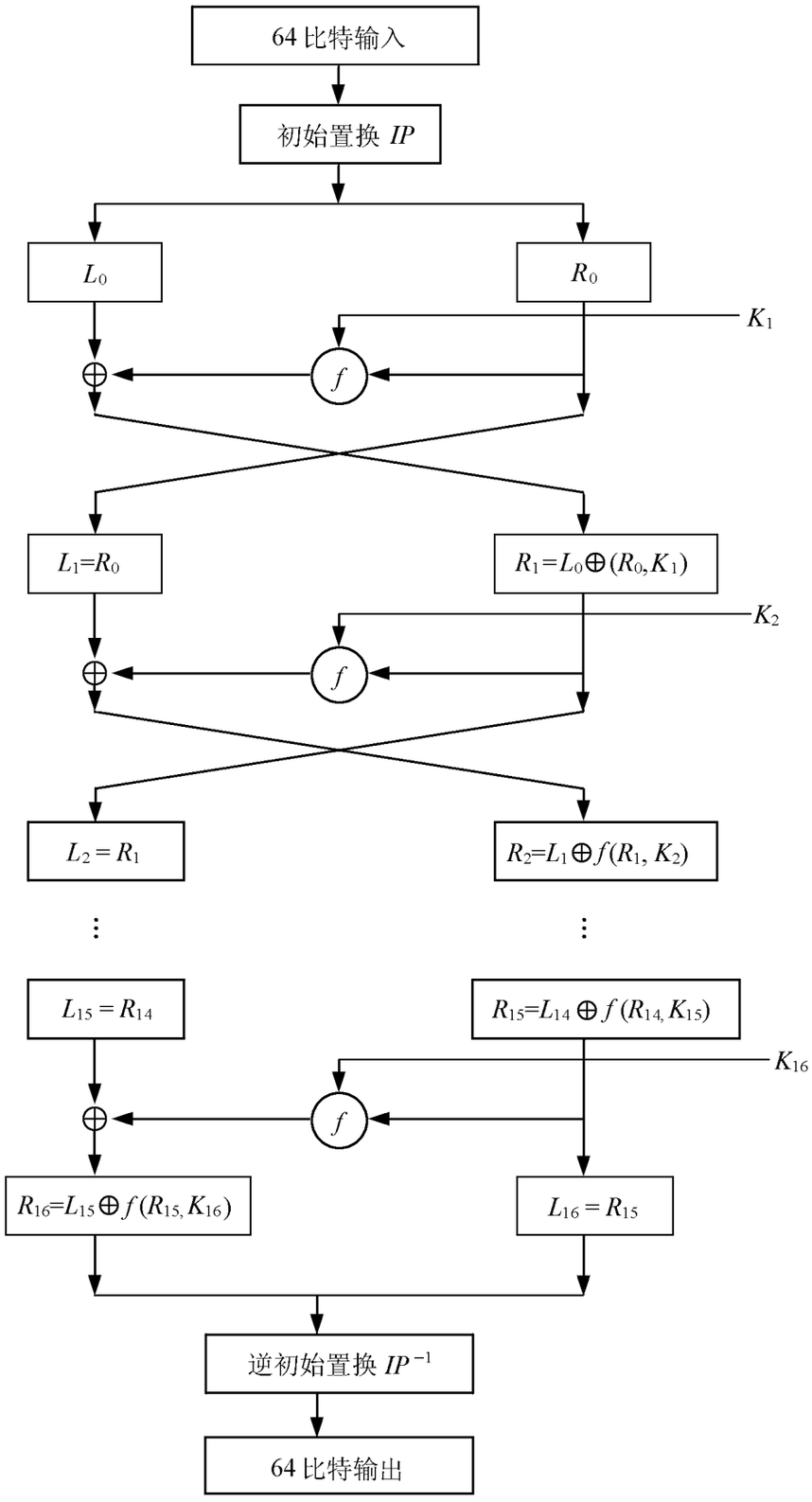


图 3.10　DES 加密算法

最后的 48 比特向量分成 8 部分，每组 6 比特分别输出到 8 个 S 盒 S_1，S_2，…，S_8 中。每个 S 盒是一个 4 行、16 列的表。盒中的每一项都是 4 比特的数。S 盒的 6 比特确定了其对应的输出在哪一行、哪一列。表 3.5 表明了 8 个 S 盒的结构。

假设 6 比特的 S 盒的输入为 b_1，b_2，b_3，b_4，b_5，b_6。b_1 和 b_6 组合成一个 2 比特数，从 0 到 3 对应着表中的一行，b_2 到 b_5 4 个比特组合成 4 比特数，从 0 到 15 对应着表的一列。例如，假设第 6 个 S 盒的输入为 110010，第一位和最后一位组合成 10(=2)，它对应着第 6 个 S 盒的第 3 行；中间 4 位组合形成 1001(=9)，它对应着同一个 S 盒的第 10 列。

第 6 个 S 盒的第 2 行、第 9 列处是数 0，则输出 0000($=0$)。

最后所有 S 盒输出的 32 比特经置换 P(见表 3.4)后，形成函数 $f(R_{i-1}, K_i)$ 的 32 比特输出(见图 3.11)。

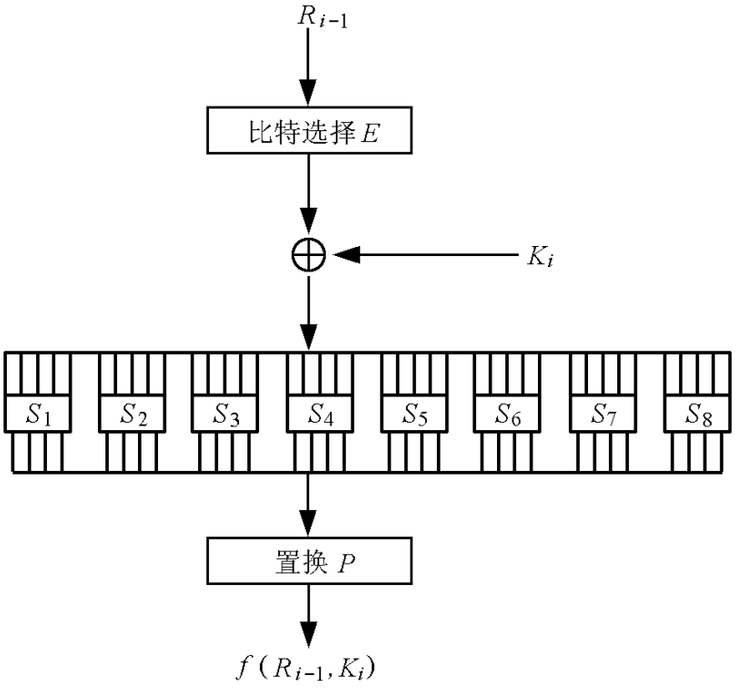


图 3.11　函数 $f(R_{i-1}, K_i)$ 的计算

表 3.5　S 盒的结构

		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
S_1	0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
	1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
	2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
	3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	3
S_2	0	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
	1	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
	2	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
	3	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9
S_3	0	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
	1	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
	2	13	6	4	9	8	15	3	0	11	1	2	12	15	10	14	7
	3	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12
S_4	0	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
	1	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
	2	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
	3	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14
S_5	0	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
	1	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
	2	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
	3	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3

S_6	0	12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
	1	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
	2	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
	3	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13
S_7	0	4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
	1	13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
	2	1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
	3	6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12
S_8	0	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
	1	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
	2	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
	3	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

2) 子密钥的产生

实际上，K 是长度为 64 的比特串，其中 56 比特是密钥，8 比特是奇偶校验位(为了检错)，奇偶校验位分布在位于 8，16，…，64 位置上。56 位密钥经过置换选择 1、循环左移、置换选择 2 等变换，产生 16 个子密钥，子密钥产生过程见图 3.12。

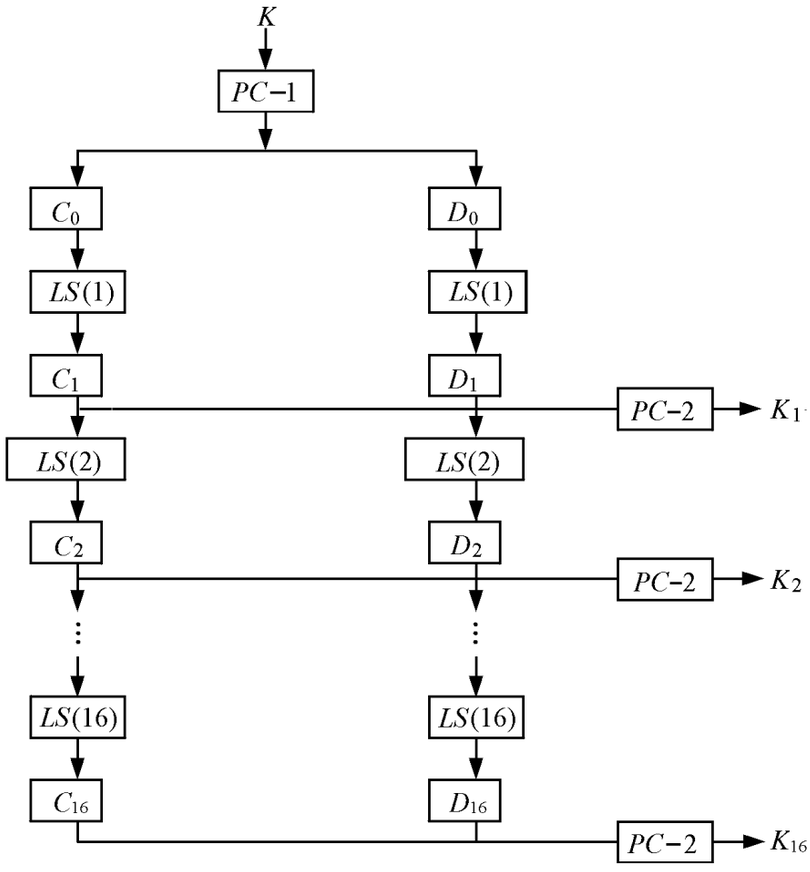


图 3.12　子密钥产生的结构图

产生每个子密钥所需的循环左移位见表 3.6。

表 3.6　循环左移位数表

圈　i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$LS(i)$	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

置换选择 1(*PC*-1)规定 C_0 的各位依次为密钥中的第 57,49,…,44,36 位,D_0 的各位依次为密钥中的第 63,55,…,12,4 位,具体见表 3 .7。置换选择 2 从 C_i 和 D_i(其 56 位)中选择出一个 48 位子密钥 K_i,其中 K_i 中的各位依次是 C_i 和 D_i 中的 14,17,…,29,32 位。具体见表 3 .8。

表 3.7 *PC*-1

57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

表 3.8 *PC*-2

14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

DES 的解密与 DES 的加密一样,只不过是子密钥的顺序相反: K_{16}～K_1。

3) 关于 DES 的实际应用

虽然 DES 的描述是相当长的,但它能以硬件或软件的方式非常有效地实现。需完成的算术运算仍为比特串的异或。扩展函数 E, S 盒、置换 IP 和 P 以及 K_1, K_2, …, K_{16} 的计算都能在一个固定时间内通过查表(以软件或电路)来实现。

DES 的一个非常重要的应用是银行交易。在银行交易中使用了美国银行协会开发的标准。DES 用于加密个人身份识别号(PIN)和通过自动取款机(ATM)进行的记账交易。票据交易所内部银行支付系统(HIPS)用 DES 来鉴别每周 1.5×10^{12} 美元的交易。

4) FEAL 和 IDEA

DES 颁布之后迅速得到了广泛应用。随着对 DES 的实际应用和深入研究,人们发现 DES 存在一些缺点,希望对 DES 进行改进或重新设计新的分组密码。新的分组密码有许多,其中最著名的是日本学者清水明宏和宫口庄司发明的 FEAL(快速加密数据算法)和我国学者来学嘉等人发明的 IDEA(国际数据加密标准)。有兴趣的读者请参阅有关参考书。

第 4 章　公开密钥密码体制

在前面介绍的传统密码体制中，加密密钥和解密密钥是一样的，任何人只要获得加密密钥就可以得到解密密钥，从而可以对用此加密密钥加密的密文进行解密，获得明文。因此在传统密码体制中，密钥是不能公开的，任何要通信的双方只有先确定密钥才能进行保密通信。

在当今信息社会中，计算机及计算机通信网络已广泛应用于社会的各个领域。利用计算机网络进行资金转移、商业谈判、采购销售等商务活动比以前更加方便快捷，而这些商务活动的信息在某种意义上就是财富，因此其信息的保密是人们迫切需要的。假如一个计算机网络有 n 个用户，那么网络就需要有 $n(n-1)/2$ 个密钥。当 n 较大时，这个数是很大的。同时为了安全的要求，通信双方要经常地更换密钥，如此大的密钥要经常地产生、分配与更换，其困难性是可想而知的，有时甚至是不可能实现的；另一方面，利用计算机网络进行商务活动，其信息的真实性也是人们迫切需要的。为了防止欺诈，通信双方必须对对方的身份、消息的真伪进行验证，有时还需要通信双方对信息进行数字签名，以便在发生纠纷时，能够提交第三者(如法院)进行仲裁。这一切都使得传统密码体制越来越不能适应计算机网络保密通信要求了，人们迫切需要寻找新的密码体制。

大家都知道，现实社会中存在一些所谓单向“街道”，沿着这个街道从 A 地到 B 地很容易，而从 B 地到 A 地却很困难。比如说一个大城市的电话号码簿，给定一个人的姓名，你可以很容易按姓氏笔画查出他的电话号码；而任意给定一个电话号码，要知道是谁的电话则很困难，有时甚至需要查阅整个电话号码本。这就是所谓的单向函数。

定义 4.1.1　函数 $f(x)$ 是单向函数，如果给定 x，求 $f(x)$ 是容易的；而给定 $f(x)$，求 x 则是困难的。这里“难”意味着即使世界上所有的计算机都用来计算这道难题，也要费很长时间。

前面介绍的整数分解、背包和离散对数的 NP - 问题就是单向函数。

1976 年，美国学者 Diffie 和 Hellman 根据单向函数的概念提出了公开密码密钥体制，引起了密码学一场革命。公开密钥密码体制从根本上克服了传统密码体制的困难，解决了密钥分配和消息认证等问题，特别适合于计算机网络系统的应用。

公开密钥密码体制简称公钥体制，其基本思想是利用求解某些数学难题的困难性，它与传统密码体制不同，用户的加密密钥与解密密钥不再相同，从加密密钥求解密密钥是非常困难的。因此用户加密密钥可以公开，登记在网络的密钥数据库中，就像把自己的电话号码公开在电话号码本上，任何人要与某个用户 U 通信，只要在公开的密钥数据库中查得用户 U 的加密密钥，用此加密密钥加密明文变成密文，将密文传送给指定用户 U，任何人如果没有解密密钥都不能恢复出明文。用户 U 可以用仅有自己知道的解密密钥对收到的密文进行解密，恢复出明文，从而完成保密通信。

在公钥体制中,加密密钥往往就是加密变换,记为 E,解密密钥往往就是解密变换,记为 D。因此,下面我们对加密(解密)密钥与加密(解密)变换不加区别。下面介绍几种常见的公钥体制。

4.1 RSA 体制和 Rabin 体制

在 Diffie 和 Hellman 提出了公开密钥密码体制的设想以后,先后由 Merkle 和 Hellman 提出了背包公钥体制,Rivest, Shamir 和 Adlemaml 联合提出简称为 RSA 的公钥体制。背包密码体制将在下一节中介绍,本节介绍 RSA 体制及其变形。

1. RSA 体制

RSA 体制是美国麻省理工学院(MIT)Rivest, Shamir 和 Adleman 于 1978 年提出来的,它是第一个成熟的、迄今为止理论上最为成功的公开密钥密码体制,它的安全性基于数论中的 Euler 定理和计算复杂性理论中的下述论断:求两个大素数的乘积是容易计算的,但要分解两个大素数的乘积,求出它们的素因子则是非常困难的,它属于 NP - 完全类。下面讨论 RSA 加、解密过程。

1) 密钥生成

(1) 随机选取两个大素数(比如 200 位十进制数) p 和 q,令 $N = pq$,随机选取两个整数 e 和 d,使得 e, d 与 $\phi(N)$ 互素,且 $ed \equiv 1 \bmod \phi(N)$;

注:$\phi(N)$ 就是第 2 章介绍的 Euler 函数。

(2) 公开 N, e,作为 E,记为 $E = (N, e)$;

(3) 保密 p, q, d 与 $\phi(N)$,作为 D,记为 $D = (p, q, d, \phi(N))$(其实 p, q 可以丢掉,但绝不能泄露)

2) 加密过程

(1) 在公开密钥数据库中,查得用户 U 得公钥:$E = (N, e)$;

(2) 将明文分组 $x = x_1 x_2 \cdots x_r$,使得每个 $x_i \leq N$, $i = 1, 2, \cdots, r$;

(3) 对每一组明文作加密变换

$$y_i = E(x_i) \equiv x_i^e \bmod N, \quad i = 1, 2, \cdots, r;$$

(4) 将密文 $y = y_1 y_2 \cdots y_r$ 传送给用户 U。

3) 解密过程

(1) 先对每一组密文作解密变换 $x_i = D(y_i) \equiv y_i^d \bmod N$;

(2) 合并分组得到明文 $x = x_1 x_2 \cdots x_r$。

下面证明解密过程是正确的:

设 x_i 与 N 互素,即 $\gcd(x_i, N) = 1$

∵ $ed \equiv 1 \bmod \phi(N)$

∴ 存在某个整数 k,使得 $ed \equiv 1 + k\phi(N)$

$$
\begin{aligned}
D(y_i) &\equiv y_i^d \bmod N \\
&\equiv x_i^{ed} \bmod N \\
&\equiv x_i^{1+k\phi(N)} \bmod N \\
&\equiv x_i \cdot x_i^{k\phi(N)} \bmod N \\
&\equiv x_i
\end{aligned}
$$

如果 X_i 与 N 不互素,也能证明

$$D(y_i) = x_i$$

因此解密过程是正确的。

例 4.1 假设 B 选择 $p = 101$ 和 $q = 113$,那么 $n = p \times q = 101 \times 113 = 11413$, $\phi(n) = 100 \times 112 = 11200$;选择 $e = 3533$,那么 $d = e^{-1} \bmod 11200 = 6597$。B 公开 $n = 11413$ 和 $e = 3533$,现假设 A 想发送密文 9226 给 B。A 计算 $9226^{3533} \bmod 11413 = 5761$,将 5761 通过公开信道传送给 B。B 收到 5761 后,进行解密,$5761^{d} \bmod 11413 = 5761^{6597} \bmod 11413 = 9226$,从而 B 获得明文 9226。

2. Rabin 体制

根据前面讨论,若存在能够分解两大素数乘积的办法,那么 RSA 体制就可以被攻破了。但是人们并没有证明要攻破 RSA 体制就一定要分解两个大素数的乘积。Rabin 基于 RSA 体制,提出了一种变形的公钥密码体制,这就是 Rabin 体制。从理论上可以证明要攻破 Rabin 体制,必须要分解两个大素数的乘积。下面介绍 Rabin 体制的加解密过程。

1) 密钥生成

(1) 随机选取两个大素数 p 与 q,令 $N = pq$,随机选取整数 $B < N$;

(2) 公开 N, B 作为 E,记为 $E = (N, B)$;

(3) 保密 p, q 作为 D,记为 $D = (p, q)$。

2) 加密过程

(1) 在公开密钥数据库中,查得用户 U 的公钥 $E = (N, B)$;

(2) 将明文分组 $x = x_1\, x_2 \cdots x_r$,使得每个 $x_i \leq N$, $i = 1, 2, \cdots, r$;

(3) 对每一组明文作加密变换, $y_i = E(x_i) \equiv x_i(x_i + B) \bmod N$;

(4) 将密文 $y = y_1\, y_2 \cdots y_r$ 传送给用户 U。

3) 解密过程

(1) 对每一组密文 y_i,求 $x_i^2 + Bx_i \equiv y_i \bmod N$ 之解 x_i;

(2) 合并分组得到明文 $x = x_1\, x_2 \cdots x_r$。

3. 素性检测

在建立 RSA 和 Rabin 密码体制时,产生大的“随机素数”是必要的,在实际应用中,所用的方法是先产生一个大的随机数,然后对此进行素性检测。

定理 4.2.1 设 p 是素数,那么

$$\frac{a}{p} \equiv a^{(p-1)/2} \bmod p$$

这里 $\frac{a}{p}$ 是 2.1 节定义的 Legendre 符号。

定理 4.2.2 设 $n>1$ 是素数,那么

$$\frac{a}{n} \equiv a^{(n-1)/2} \bmod n$$

另一方面,如果 n 是合数,则至多有一半的整数 $a(1\leq a\leq n-1)$ 满足

$$\frac{a}{n} \equiv a^{(n-1)/2} \bmod n$$

根据这两个定理,我们就可以对一个随机数进行素性检测了。

素性检测

1) 随机选取一个整数 $a,(1\leq a\leq n-1)$;

2) 计算 $\frac{a}{n}$;

3) 如果 $\frac{a}{n} \equiv a^{(n-1)/2} \bmod n$,则回到步骤 1,否则认为 n 是合数,退出;

4) 重复步骤 1,2,3 共 t 次(t 任意取),如果每次都有

$$\frac{a}{n} \equiv a^{(n-1)/2} \bmod n$$

则一般认为 n 是素数。

4.2 背包体制

背包公钥体制是 1978 年由 Merkle 和 Hellmam 基于求解背包问题的困难性而提出的一个公开密钥密码体制。根据 2.3 节的介绍,背包问题属于 NP-完全问题,是比较困难的问题。虽然背包问题属于 NP-完全问题,但不是任意背包问题都是很困难的,有时也是很容易的。例如,超递增背包序列就很容易求解。

定义 4.3.1 正整数序列 $a_1, a_2, \cdots, a_n$ 称为超递增的,若对任何 $1\leq l\leq n-1$,有

$$\sum_{i=1}^{l} a_i < a_{l+1}$$

例 4.2 $(1,2,4,8,16)$就是一个超递增序列。

定理 4.3.1 由超递增序列 $a_1, a_2, \cdots, a_n$ 及 S'确定的超递增背包问题是容易求解的。

下面讨论背包体制的加密、解密过程:

1. 密钥生成

1) 随机选取一超递增序列$(e_1, e_2, \cdots, e_n)$,作为用户的秘密密钥,记为 $D=(e_1, e_2, \cdots, e_n)$

2) 选取一对大的且互素的数 w, N,并把背包序列$(e_1, e_2, \cdots, e_n)$变为困难的背包

序列

$$T(e_i) \equiv we_i \bmod N$$

3) 将$(T(e_1), T(e_2), \cdots T(e_n))$公开作为 E,记为 $E = (T(e_1), T(e_2), \cdots, T(e_n))$

2. 加密过程

1) 在公开密钥数据库中查得用户 U 的公开密钥

$$E = (T(e_1), T(e_2), \cdots, T(e_n))$$

2) 将明文表示成二元序列,并适当分组 $x = x_1 x_2 \cdots x_r$,每组长 n 比特

3) 对每一组明文作加密变换

$$y_i = E(x_i) = \sum_{j=1}^{n} x_{ij} T(e_j), \quad i = 1, 2, \cdots, r$$

4) 将密文 $y = y_1 y_2 \cdots y_r$ 传送给用户 U

3. 解密过程

1) 计算 $y_i' \equiv w^{-1} y_i \bmod N$

2) 按照超递增向量序列$(e_1, e_2, \cdots, e_n)$,从 y_i' 还原 x_i

3) 合并分组得到明文 $x = x_1 x_2 \cdots x_r$

例 4.3 设 $n = 10$,考虑超递增序列

$$A = (103, 107, 211, 430, 863, 1718, 3449, 6907, 13807, 27610)$$

选取 $N = 55207$, $W = 25236$,则 $W^{-1} = 1061$。将超递增序列向量 A 变为困难的背包向量。

$$B = (4579, 50316, 24924, 30908, 27110, 17953, 32732, 16553, 22075, 53620)$$

某人要传送明文 $x = 1010110100$ 给用户 U,他首先计算

$$y = E(x) = \sum_{i=1}^{10} x_i B_i = 4579 + 24924 + 27110 + 17953 + 16553 = 9119$$

然后将 91119 传递给用户 U,用户 U 收到 91119 后,计算

$$y' \equiv w^{-1} y \bmod N \equiv 1061 \times 91119 \bmod 55207 = 9802$$

然后,解超递增背包问题,得 $x = 10110110100$。这样用户 U 就还原了明文。

Merkle 和 Hellman 提出背包体制时曾认为它是安全的、不可攻破的,但 5 年之后,Shamir 完全破译了背包体制,目前背包体制一般不再使用了。

4.3 EIG amal 体制

EIG amal 公钥体制是基于离散对数问题的难解性,根据 2.3 节的介绍,离散对数问题是属于 NP - 完全类。下面介绍 EIG amal 体制。

随机选取一个大素数 p(200 位十进制数),选一个数 α(模 p 的本原根)把 p, α 对每个用户公开。

1. 密钥生成

1) 随机选取一整数 x 作为用户的秘密密钥,记为 $D=(x)$;

2) 计算 $y\equiv a^x \bmod p$;

3) 将 $E=(y)$公开,作为用户的公开密钥。

2. 加密过程

1) 在公开密钥数据库中查得用户的公开密钥: $E=(y)$;

2) 随机地在 0 与 $p-1$ 之间取一整数 k;

3) 计算 $K\equiv y^k \bmod p$, $c_1\equiv a^k \bmod p$, $c_2\equiv Km \bmod p$;

4) 取(c_1, c_2)作为 m 的密文传送给用户 U。

3. 解密过程

1) 计算 $K\equiv c_1^x \bmod p$;

2) 计算 $m\equiv c_2 K^{-1} \bmod p$。

4.4 概率加密体制

上面介绍的三种公钥体制都是确定型公钥体制,所谓确定型公钥体制是指任一明文的密钥都是由公钥唯一确定的。这种确定型公钥体制存在很多缺陷,不能达到严格的安全保密要求。例如,股票市场的"买进"与"抛出"是非常有价值的信息,某人可以用某些大户的加密函数 D,对这些有价值的信息预先加密并保存,则他一旦获得某些大户的密文后,就可以直接在所存储的密文中进行查找,从而求得相应的明文,获得有价值的信息。

1982 年,Goldwasser 和 Micali 等人提出了概率加密的概念,较好地克服了确定型公钥密码体制的缺陷,下面将介绍有关方法。

1. GM 体制

GM 体制是由 Goldwasser 和 Micali 发明的,下面介绍详细的加密、解密过程。

1) 密钥生成

(1) 随机选择两个 k 比特的素数 p, q,并令 $N=pq$。p, q 作用户产生的秘密密钥,记为 $D=(p, q)$;

(2) 随机选择一个与 N 互素的数 y,使得 y 是 N 的二次非剩余;

(3) 将(N, y)作为用户的公开密钥,记为 $E=(N, y)$。

2) 加密过程

(1) 在公开密钥数据库中,查得用户 U 的公开密钥: $E=(N, y)$;

(2) 将明文 m 表示成二元序列 $m=m_1 m_2 \cdots m_r$;

(3) 随机送一个小于 N 且与 N 互素的数 x,计算

$$c_i = E(m_i) = \begin{cases} yx^2 \bmod N, & \text{如果 } m_1 = 1 \\ x^2 \bmod N, & \text{如果 } m_1 = 0 \end{cases}$$

(4) 将密文 $c = c_1 c_2 \cdots c_r$ 传送给用户 U。

3) 解密过程

(1) 对每一个密文 c_i 如果 c_i 是 N 的二次剩余,则 $m_i = 0$,否则 $m_i = 1$

(2) 合并分组得到明文 $m = m_1 m_2 \cdots m_r$。

2. BBS 体制

BBS 体制是由 L .Blum, M .Blum 和 M .Shub 等人于 1982 年发明的,下面介绍其加、解密过程。

1) 密钥生成

选择一对互异的素数 p, q,使得 $p \equiv q \equiv 3 \bmod 4$;

(1) 令 $N = pq$,将 N 公开,记为 $E = (N)$;

(2) 将 (p, q) 保密,记为 $D = (p, q)$。

2) 加密过程

(1) 随机选择一个小于 N 且与 N 互素的数 x;

(2) 令 $x_0 \equiv x^2 \bmod N$;

(3) 对 $i = 1, 2, \cdots, r$ 执行(4),(5);

(4) $x_i \equiv x_{i-1}^2 \bmod N$;

(5) $b_i = x_i$ 的最后一个比特;

(6) $x_{r+1} \equiv x_r^2 \bmod N$;

(7) 密文 $E(m) = (m \quad b, x_{r+1})$ 其中 $b = b_1 b_2 \cdots b_r$。

3) 解密过程

(1) 用户利用自己的秘密密钥快速地由 x_{r+1} 恢复出 $x_r, x_{r-1}, \cdots, x_1$,因此求得 $b = b_1 b_2 \cdots b_r$;

(2) 将 $b = b_1 b_2 \cdots b_r$ 与密文 $E(m)$ 的第一个分量作模 2 加运算,求得明文 m。

除了前面介绍的常见的公钥体制外,还有许多公开密码密钥体制,散见于各个参考文献中。例如,利用二次域的 William 体制,基于编码理论的 McEliece 公钥体制,基于语言理论的公钥体制,特别值得一提的是,我国学者陶仁骥等人提出了一种基于有限状态自动机的可逆性理论的公开密码密钥体制,这个体制具有一些显著的优点,得到国际上广泛关注,有兴趣的读者可参阅有关参考书。

第 5 章　信息安全与保密技术

本章首先介绍操作系统和数据库的安全与保密问题,然后讨论 DSA 数字签名算法,并且给出使用 DSA 生成、验证签名的例子,最后讲述智能卡和电子数据传输的安全保密问题。

5.1　操作系统的安全与保密

操作系统是重要的系统软件之一,目前的多数操作系统支持多道程序设计和资源的共享,能够对计算机的硬件资源和软件资源实行统一的管理和控制。正因为操作系统有如此重要的功能,因而也容易受到攻击。因此,需要研究操作系统的安全性问题,一方面要研究如何设计和实现一个安全操作系统,另一方面要研究安全操作系统能够为用户提供的各种保护措施。

1. 安全操作系统设计

安全操作系统需要处理许多任务,要控制和管理系统中数据的存取、程序的运行和外部设备的工作,还要使负载最小以至于不减慢计算的速度,因此,操作系统本身必须是安全的。实现操作系统安全性的任务,实质上增加了设计操作系统的难度。

在设计安全操作系统时,要讨论三个性质: 隔离性,通过隔离技术,一个操作系统支持用户领域的共享和隔离;核心设计,这是保证安全性的有效方式;层或环结构设计。

1) 安全操作系统设计原理

萨尔哲(Saltzer)和史克罗德(Schroder)提出了安全操作系统设计原理。

(1) 最小特权　为使无意的或恶意的攻击所造成的损失达到最低限度,每个用户和程序必须尽可能地使用最小特权。

(2) 经济性　为了便于能得到验证和正确的执行,设计的操作系统应该是足够小和足够简单的,并且易于理解。

(3) 开放系统设计　保护机构应该是公开的,不取决于公开者的无知,而只取决于相对少的关键项的安全性,如口令表,开放系统设计对广泛的公共的仔细检查很有用。

(4) 以许可为基础　存取时否认,排除违约条件,保守的设计者能识别可存取的一些项,而不是不可存取的那些项。

(5) 公用机构最少　由于多个用户共享存取目标常常会出现意想不到的潜在的信息通道。需要通过物理隔离的硬件或逻辑隔离的虚拟机使用户隔离,以减少共享所产生的错误。

(6) 有效性　存取控制机构应该是有效的,操作系统中的每一次存取都必须受到

控制。

(7) 完全协调　每次存取应该经过检查;该机构必须有高效率;用户应能够掌握;三者应该协调。

(8) 方便性　使用的存取控制机构,要使用户在心理上容易接受且乐于使用。

2) 基本的多程序操作系统特征

安全性能体现在整个操作系统的设计和结构中,这包括两方面含义: 第一,在操作系统设计的每个部分都应该考虑安全性,当设计某一部分时,应该检查它提供或实现的安全度。第二,安全性应该体现在整个系统中,在操作系统初始设计时就应考虑安全性,在已经设计好而没有考虑安全性的操作系统中再附加安全性能是很难做到的,操作系统初始设计就应该考虑安全性。

操作系统实现以下几个安全性能:

(1) 用户识别: 操作系统应该对有存取要求的每个用户进行身份识别,并保证每个用户都是它所支持的用户,最普通的识别机构就是口令比较。

(2) 存储器保护: 每个程序在指定的存储器空间运行,非法用户不能对其进行访问,这种保护控制了用户本身不能存取非授权的程序空间。

(3) 通用目标分配和存取控制:操作系统应该给用户提供通用目标,像允许并行和同步的机构。但是应该控制使用这些目标,避免一个用户对其他用户有负作用和不良影响。

(4) 文件和 I/O 设备存取控制: 操作系统应该保护用户和系统文件,防止未授权的用户进行存取。同样地,也应该保护 I/O 设备。

(5) 保证公平服务: 所有用户都希望得到中央处理器和系统提供的其他服务,不能让用户无限期地等待服务,让硬件时钟和调度规则结合起来使用,以提供公平服务。

(6) 进程通信和同步: 正在执行的过程有时需要和其他过程进程进行通信或同步存取共享的资源,操作系统作为过程之间的桥梁,提供这些服务,还提供和其他过程的异步通信。

3) 隔离性设计

隔离是指采用一定措施使系统某一部分的问题不影响其他的部分。也就是说,系统的一部分出了问题,如软件出错或硬件出现故障,限制在尽可能小的范围内,使之造成的损失最小。设计的思想是把一个大系统分割成若干个互不相交的小系统,对一个任务来讲,由几个独立的小系统各自独立的完成自己的任务,且每两部分之间都有“保护林”。

隔离性设计有四种隔离方式: 物理隔离、暂时隔离、密码技术隔离和逻辑隔离。在物理隔离中,过程处理使用了不同的硬件设备,如敏感的计算任务在指定的系统中执行,非敏感的计算任务在开放系统中执行。暂时隔离指的是不同的运行时间运行不同的过程。例如,军事系统在 8 点到中午时间执行非敏感任务,只在中午到下午 5 点执行敏感任务。密码技术隔离是将加密技术用于隔离,使未授权用户不能以读的形式存取敏感数据。逻辑隔离也称分离,例如,监控器隔离用户对象。安全操作系统通常是所有的这些隔离形式的组合。

4) 核心设计

核心是操作系统中完成最低级功能的部分。在标准的操作系统设计中,核心完成许

多操作,如同步进程通信、信息传递和中断处理等。

安全核心负责完成全部操作系统的安全机构。安全核心在硬件、操作系统和其他部分计算机系统中提供接口。

安全核心的设计和用途在一定程度上取决于设计策略,一个核心可以当作额外的操作系统,也可以当作整个操作系统的一部分来设计。

5) 层结构设计

一个核心操作系统至少由四层组成: 硬件、核心、操作系统和用户。每一层本身也包括子层。例如,在用户层,通常有标准系统程序,像数据库管理器或用户接口,由隔离的安全层组成。

可以用一系列同心环来描述安全操作系统特点。其中,在最内层进行最敏感的操作,一个过程的可信性和存取权由到中心的接近程度来决定,越可信的过程越接近中心,这样的一个系统可用图 5.1 表示。

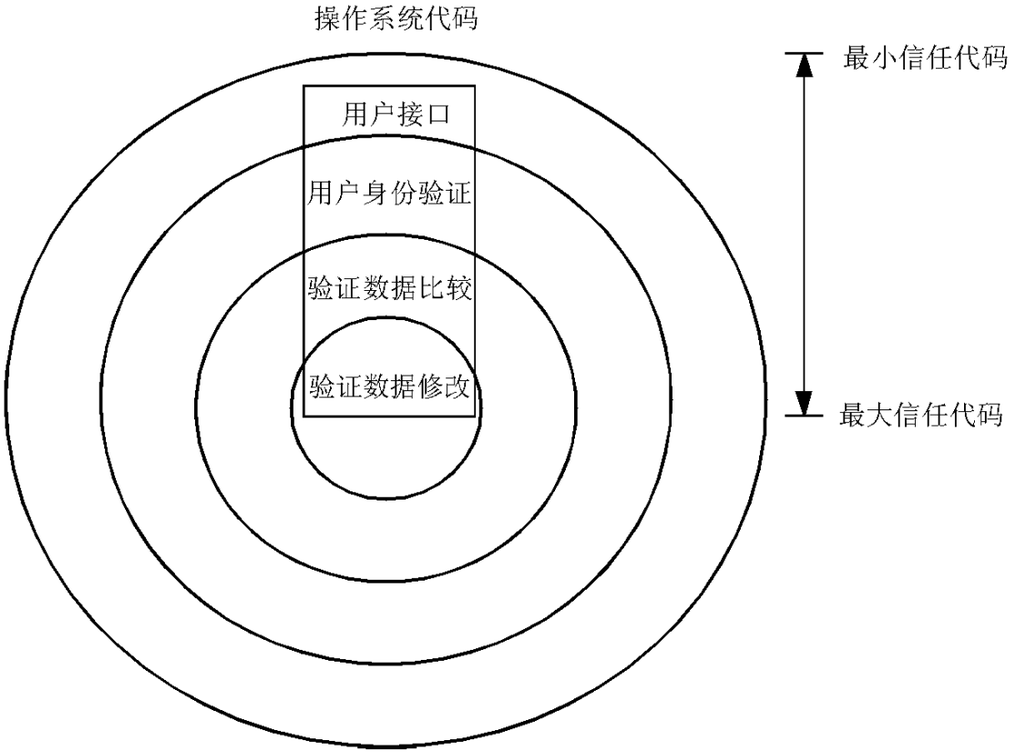


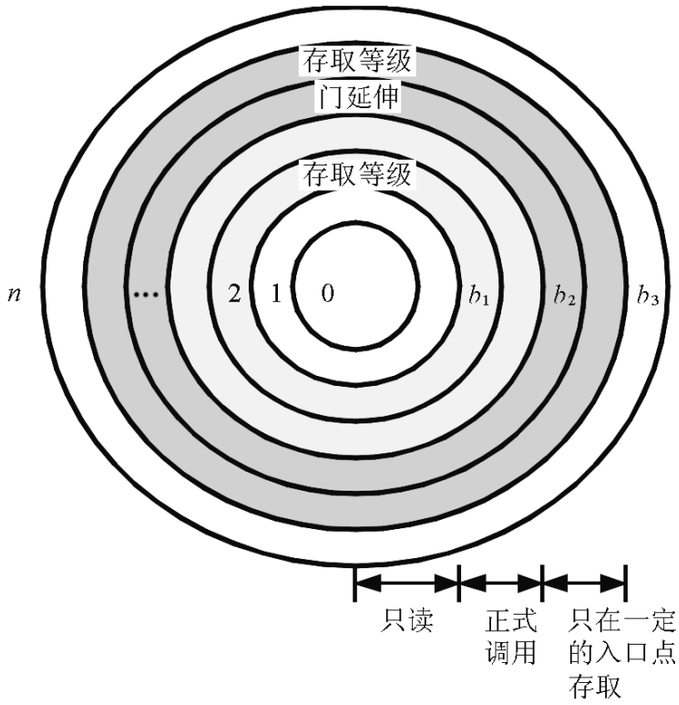
图 5.1 不同层次上的模块操作

在这种设计中,使其在安全核心外部完成一些保护功能。例如,用户识别包括存取口令表、要求用户提供口令、验证口令正确性等。

层状设计策略被认为是一种较好的操作系统设计策略。每一层都把更中心的层作为服务程序,每层都给外层用户提供一定等级的服务。这样,即使剥去一层,还有一个功能上较少的逻辑上完整的系统。

6) 环结构设计

环定义了一个过程的存取权。环标记着数字,从 0 开始,核心标记为 0,环好像是围绕操作系统硬件的同心带。每一个过程都在特殊的环等级上运行(图 5.2)。比较可信的过程在标记有低数字的环上运行。环是交互重叠的,以便在 I 级上的运行包括所有的 J 级环上的特权,这里 $I < J$,环数越低,过程的存取权越大,其操作系统受的保护越少。



$b_1 < b_2 < b_3$，(b_1, b_2)为存取等级，(b_2, b_3)为调用等级或门延伸，从 b_1 到 b_2 的范围是可自由地对程序段进行存取的过程的环集。

图 5.2　环等级

2. 操作系统保护的对象及方法

一个安全的操作系统应该对存储器的存取和对象(如文件、目录等)两方面提供有效的保护服务。

1) 对存储器的保护

多道程序设计最显著的问题是如何防止一个程序影响其他程序的存储空间。系统可以通过硬件,对存储器的存放使用提供保护。而这种保护无需增加操作系统的任何开销。常用的保护方法有栅栏保护、再定位保护、基址/界限寄存器保护、标志位保护、分段保护和分页保护等。

(1) 栅栏保护

在单用户系统中,可以采用一种最简单的存储器保护方法——栅栏法,以避免伪用户程序破坏操作系统的常驻程序。下面介绍两种常用的实现方法。

这种保护的一种实现方法是把栅栏看成是优先定义的存储器地址。操作系统以此地址为界限,驻在存储器一端,而用户使用存储器另一端,这种情况如图 5.3 所示。但这种实现方法是很受限制的,原因是操作系统占用了固定量的存储空间。如果需要的空间量少于这个固定量,那么一部分空间会浪费掉,而操作系统不能越过栅栏界限。

另一种实现方法是使用寄存器。这种寄存器也称作栅栏寄存器,它包括操作系统的末端地址,每次一个用户程序为修改数据生成一个地址,这个地址自动地与栅栏地址比较,如果这个地址是操作系统域地址,则会引起错误。

但是栅栏保护方法不能保证一个用户不受其他用户的侵扰。同时,也不能识别一定的程序域是否是无损的,或不受侵扰的。

(2) 再定位保护

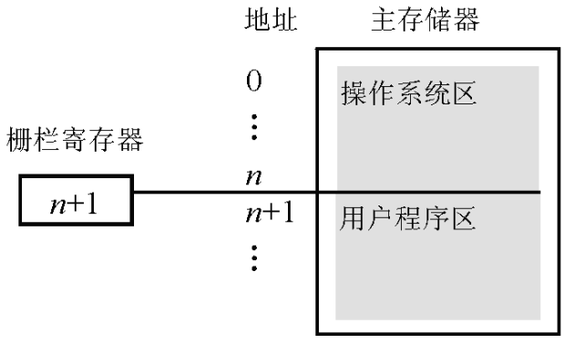


图 5.3　把栅栏看成是优先定义的存储器地址的一种用户保护服务方法

如果可以假定操作系统有固定的大小,那么程序员可以假定从一个常数地址开始存储,他可以写入程序代码,这很容易确定程序中任意目标的地址,并且不可能改变这个初始地址。

再定位是写入程序的一个过程,好像程序从 0 地址开始存储,改变了影响存储器中程序存储的实际地址的所有地址,对地址起到了保护作用。在许多例子中,这只需要对每个程序地址增加一个不变的再定位因子,再定位因子是给每个程序分配的存储器的开始地址。

再定位保护方法中使用栅栏寄存器优点更为突出。栅栏寄存器可以是硬件再定位装置。对每个程序地址来说,栅栏寄存器的内容可以增加。这样不仅定位了地址,而且也保证了不可能对低于栅栏地址位置的存储器进行存取。

(3) 基本/ 界限寄存器保护

如果把一个变化的栅栏寄存器称作基本寄存器,那么它只提供一个基本地址或开始地址,没有提供高地址。为此应该再增加一个寄存器,如图 5.4 所示,增加的寄存器称作界限寄存器,用于存储程序的最高地址或结束地址。这样,基本/ 界限寄存器存储了一个用户存储区的界限信息(开始地址和结束地址)。因此要求每个程序地址高于每个基本地址,而低于界限地址。

这样的保护措施可以防止用户自己无意地访问其他用户的存储区, 也可以阻止别的

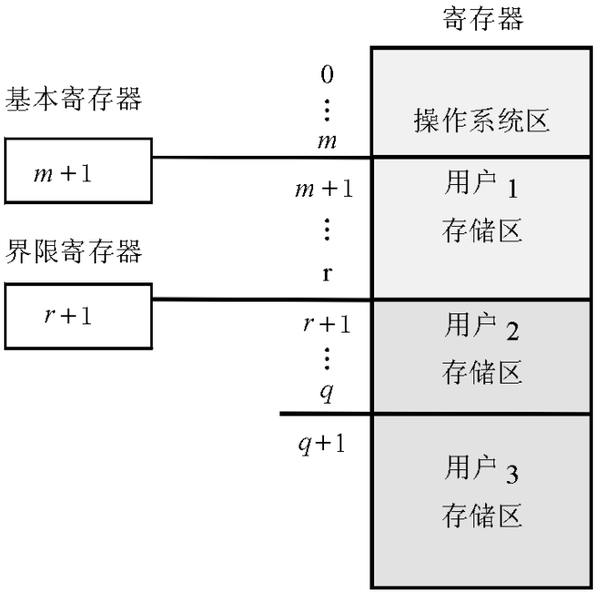


图 5.4　基本/ 界限寄存器对

用户非法访问自己的存储区。当一个用户程序执行到另一用户程序时,操作系统必须改变基本寄存器和界限寄存器的内容以反映用户的真实地址区域。

使用这种保护方法应注意:对于基本/界限寄存器中的内容必须加以保护。防止随意修改;对于多用户、多道程序的系统,必须提供多对基本/界限寄存器。如果数量不够,那么在执行不同的用户程序之前需要修改寄存器的内容,这很耗费时间。另外,这种方法只适用于保护用户连续的情况。

(4) 标志位保护

标志位保护方法适用于字一级的保护,它是在每个字前面设置一个或多个标志位,因此标志对这个字的存取权。每次发出指令对存储器进行存取,都要对标志位进行检查。标志位如表 5.1 所示。对一部分存储内容保护成只执行,另一部分保护成只读,其他部分保护成只写。两部分相邻的存储器可以有不同的存取权。使用标志位,可以把数据(如数字、字符、地址或指针等)分成不同的保护等级,也可以把数据域保护成仅有特权指令存取。

表 5.1 标志结构

标志	存储字
写	0138
读	0002
读/写	—
执行	0001
读/写/执行	0023

(5) 分段保护

分段保护是把一个程序分成许多块,这些块是分离的,且有不同的存取权,每个块都有逻辑完整性。例如,一个段可以是单一过程的码,也可以是队列数据,也可以是某个特殊模块使用的区域的数值集合。

从逻辑上讲,程序员可以把程序看作是段的集合。段可以分别地重新分配位置,允许任意有效的存储位置来存储段,段的逻辑位置与物理位置如图 5.5 所示。

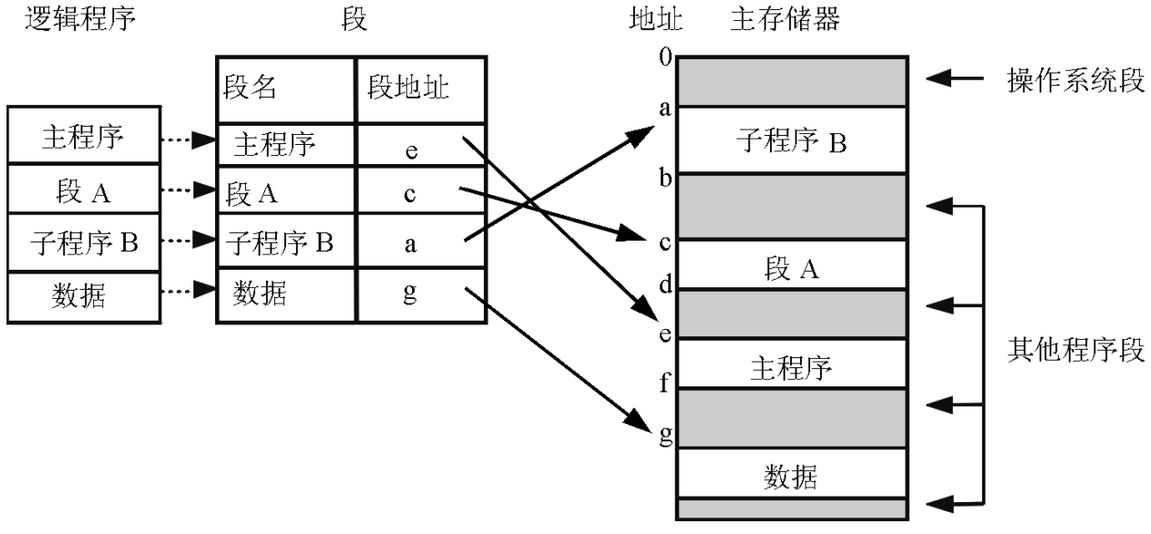


图 5.5 段的逻辑与物理表达

给用户的程序段分配不同的保护等级是可能的。软件和硬件组合使用来保护段是必要的,把一定的段和一定的保护等级联系在一起。每次对段存取要进行保护检查。例如,第一段可以只写数据,第二个段只执行,第三个段只读。

(6) 分页保护

和段一样,每个地址都分成页和偏移两部分。程序分成了大小相等的块,这些块称作“页”,同时存储器也分成同样大小的空间存放页。每个地址的转换规则与段的操作相同,保管一张页转换表(用户页数和它们在存储器中的真实地址两列),如图 5.6 所示。

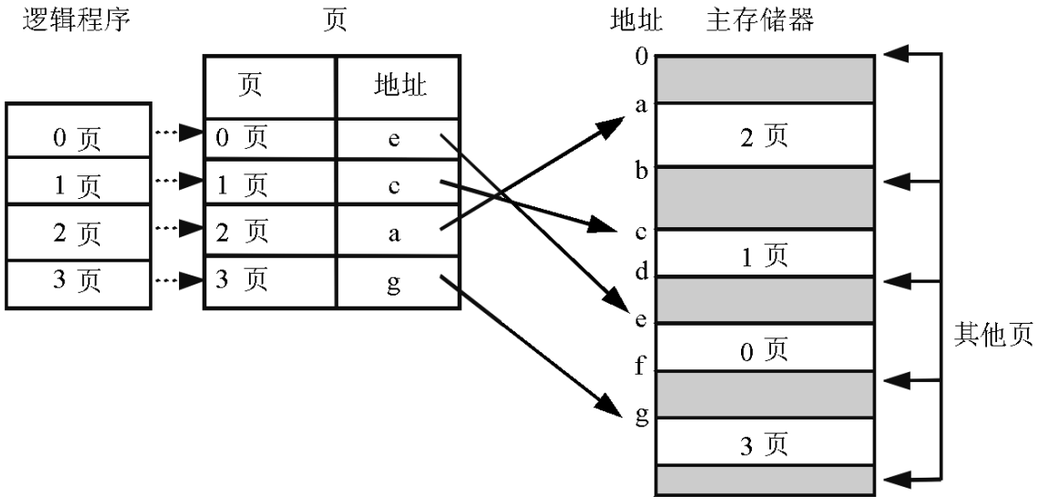


图 5.6　页转换示意图

2) 对文件目录的保护

每个文件都有一个特殊的拥有者,拥有者有主要的存取权。每个用户都有一个文件目录,列出了能存取的所有文件目录。很明显,因为写文件目录可以伪造存取权,所以不允许任何用户都可以写文件目录,应由操作系统统一保护所有的文件目录。一些常用的存取权是读、写、执行。文件目录如图 5.7 所示。

3. 访问控制

用户对对象有何种存取权限,可以用存取控制表和存取控制矩阵来表示。

1) 存取控制表

存取控制表是对通用对象保护的另一种表达方式。一个对象有一个表,用来指出主体对对象有哪些存取权。图 5.8 给出了这样的示意图。

2) 存取控制矩阵

存取控制矩阵是一个用户主体对对象有何种存取权限的矩阵,可用 $A=(a_{ij})$, $i=1,2,\cdots,N$, $j=1,2,\cdots,M$。矩阵的行 i 代表用户主体,列 j 代表对象,矩阵的元素 a_{ij} 代表某个用户主体 i 对对象 j 的存取权限,即读、写、执行或无。例如,图 5.8 中,用户 1～4 对对象 1～5(HELP,ITEMBANK,ENCRYPT,DECRYPT,DRAW)的存取权限可表示为

$$A=(a_{ij})=\begin{matrix} R & & W,E \\ R,W,E & R,W,E & & R,W,E & R \\ W & R,E & & & W \\ E & & & & \end{matrix}$$

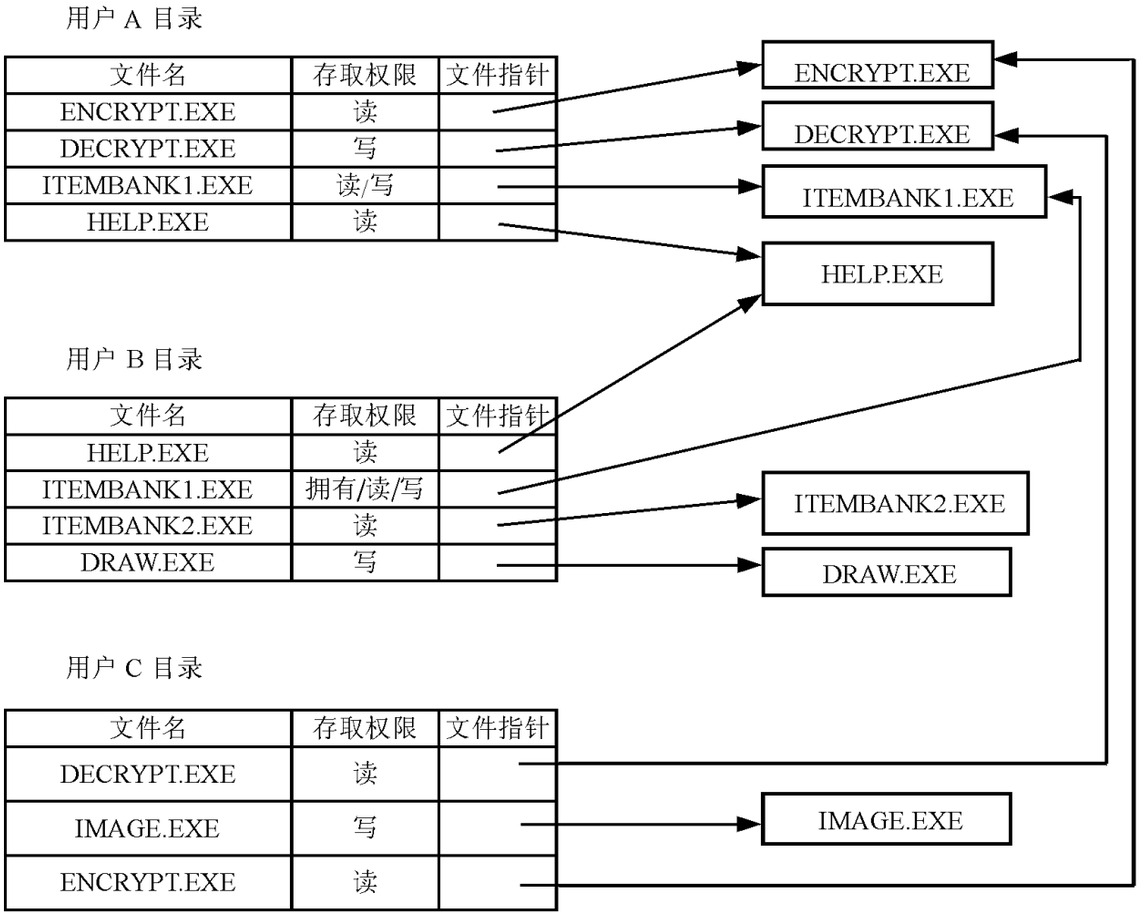


图 5.7　文件目录存取示意图

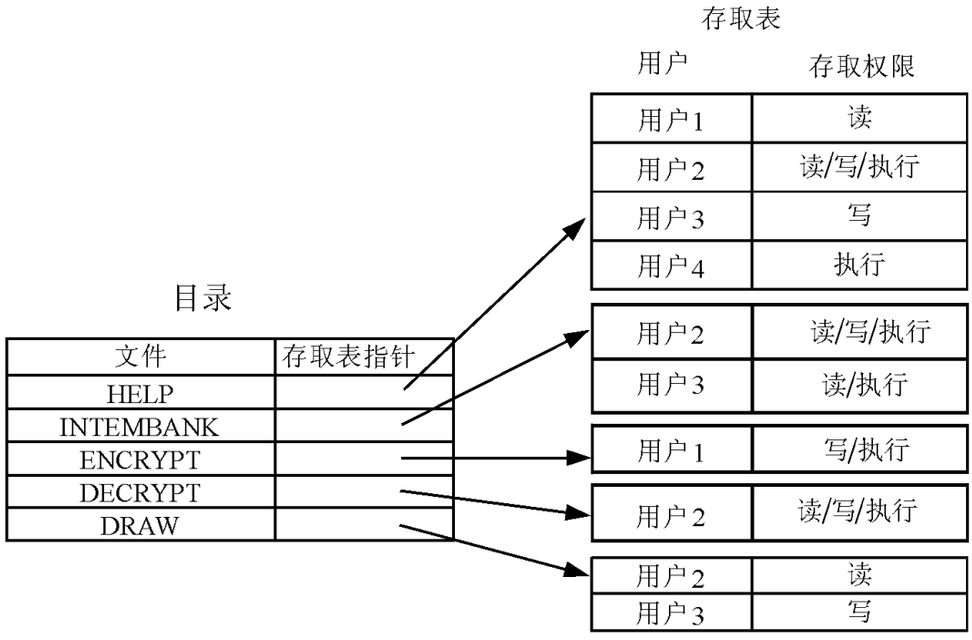


图 5.8　存取控制表

其中，*R*，*W*，*E*，　分别表示存取权限为读、写、执行、无。

4. 基于口令的用户认证

用户是否可以安全地进入某个计算机系统？可以通过验证用户输入的口令来实现。只有那些能输入正确口令的用户才能合法地进入。因此，下面就讨论有关基于口令的用

户认证问题,包括口令的种类及其选择准则、验证、口令文件的加密等。

1) 口令

口令是用户和系统间相互认可的码。口令有时由用户选择,而有时由系统统一分配。口令的长度和形式也随系统的不同而不同。

口令的使用是很直接的。系统要求用户输入口令,如果口令对了,那么用户得到了系统的证实;如果口令不对,那么用户不能被证实,这时系统要求用户再输入口令。口令本身是不安全的,它可能会受到攻击。

2) 口令验证

验证需要知道名称和口令,假定一个旁观者(包括攻击者)对系统什么也不知道,他可能试图按照下列方式攻击存取系统(例子中大写字母是系统信息,小写字母是用户的回答)。

```
WELCOME TO THE ITEM BANK SYSTEM
ENTER USER NAME: wen
INVALID USER NAME—UNKNOWN USER
ENTER USER NAME: wln
```

攻击者发现"wen"不是授权用户的名称。他可以试其他的常用名称,以建立一个授权用户名称清单。

3) 口令文件的加密

比较安全的策略是把口令表加密。加密后攻击者不能读和使用口令。两种常用的加密方法是传统的加密方法和单向函数方法。

传统的加密方法中,把整个口令表加密,或只把口令这一列加密,当接收用户的口令时,把存储的口令解密,然后比较两个口令。

较安全的策略是使用单向函数加密。有一个加密函数,使加密相对容易,使解密很难进行。例如,函数 X 容易计算,而反函数不易计算。口令表中的口令以加密的形式存储,当用户输入口令时,口令也被加密,然后比较加密后的口令。如果两者相同,那么证实是成功的。大部分安全加密算法要求:不允许两个不同的口令加密成相同的密文。单向函数加密过程如图 5.9 所示。

注册名	口令字密文
李校春	hamatailo
李桂花	kandohana
冯波	fengbo
王洪	wanghong
张博	zhangbo
李研	liyan

口令文件的明文形式

注册名	口令字密文
李校春	# @326rtuw
李桂花	( )hdghdshd
冯波	+ - uytuyft
王洪	! @ # # $ fgx
张博	dfdgsfg # $ # $
李研	# # S # $ S VCXBFS

口令文件的密文形式

图 5.9 口令文件的单向加密

4) 口令选择标准

口令应该是很难进行猜测并且很难用穷举法确定的,口令选择标准包括以下几点:

(1) 使用比英文字母 A～Z 更多的符号。如果口令只从 A～Z 的字母中选择,那么只有 26 种可能性,把数字(0～9)加进去,可能性扩大到 36 种,使用大写字母、小写字母、再加上数字,可能性就有 62 种,尽管这种变化似乎很小,当测试口令空间时,影响却是很大的。如果口令是由字母组成的,且口令长度为 6,那么测试所有的口令需要 100 小时,如果口令是由大写字母、小写字母、数字组成的 6 个符号,那么测试所有的口令大约需要两年时间。100 小时还可以测试,如果需要两年测试,攻击的兴趣就大大减小了。

(2) 选择长的口令。选择长的口令会增大猜测的难度,但也增加了记忆难度。

(3) 避免使用有特殊意义的口令,例如,实际的名称或单字。

(4) 不要写出口令,而且经常规则地变化口令,不要将口令告诉别人。

5) 一次性口令

一次性口令是每次使用后口令内容都发生变化的口令。系统不是给用户分配一个静态口令,而是分配一个静态数学函数,系统给函数提供变量值,用户计算和返回函数值。这个系统也称作问答。通常,一次性口令函数比较简单,但是,用于网络中身份证实的函数相对来说要复杂一些。

一次性口令对身份证实是很安全的,因为窃听到的口令没有用。一次性口令的用途受算法的复杂性限制,口令发生器类似于袖珍计算器,它能实现更复杂的函数,但这个发生器可能丢失。

5. 常用操作系统和工具软件的安全保护特例

1) DOS 系统安全命令

DOS 系统是常用的操作系统,下列安全命令可以提高其安全性。

(1) COPY, XCOPY 和 BACKUP

使用 COPY, XCOPY 和 BACKUP 可以从一个磁盘归档一个或多个文件到另一种介质(如软盘、磁带、数字音频磁带、光盘)上,进行备份。

(2) ATTRIB

ATTRIB 命令给数据文件额外增加一层保护。例如,利用 ATTRIB 可以将一个文件(如 TEX .EXE) 指定为只读(read-only)文件,其操作是 ATTRIB TEX .EXE + R,也可以指定一个文件为隐含(hidden)文件,不经意者不易看见。但下列情况下除外:使用 DIR * .EXE/ A 命令或 Windows 3 .x 的文件管理器 File Manager 或 Windows 95 的资源管理器 Explorer 能显示所有的文件目录。如果不能保护桌面,入侵者就可以使用 ATTRIB 命令去除这些属性。

(3) RENAME

使用 RENAME 命令更改文件名字,用来隐藏敏感的文件。

(4) FDISK

使用 FDISK 命令将磁盘分段划分成两个或多个逻辑盘,然后用口令保护一个分区。

(5) CHKDSK

用 CHKDSK 命令运行一个定期的报告,可以确定内存的有效部分是否被可疑地占用(一个潜在的病毒)或者程序是否突然占据更多内存(也是一个潜在的病毒)。

(6) FC

用 FC 命令比较磁盘上的文件,以确定是否有人更换或修改过文件,或是否受病毒感染。

2) 提高 Windows 95 安全的方法

Windows 95 是目前较为流行的操作系统,下面方法可以提供安全性。

(1) 删除带扩展名 .pwl 的文件。Windows 95 将口令存储在一个 .pwl 的文件中,为了使自己的口令对其他应用程序安全,打开 Windows 文件夹,找到 .pwl 文件并删除掉,不需将口令存入硬盘,以免他人窃取。

(2) 为避免设置好的配置被别人修改,可以不让某些图标出现在控制面板上。编辑 Windows 目录中的 CONTROL .INI 文件,找到[donot load]段,对每个想使其失效的图标加上〈cpl file〉= no。并用相关的 .cpl 文件代替 cpl file。

(3) 在用户使用共享级安全措施时,为了保护用户自己的目录,可以在保护的目录名的末尾加一个 $,将其从网络邻居浏览表中隐藏起来。

(4) 为避免未经授权的访问,使用键盘锁,或使开机口令失效,从而在物理上保护系统。

(5) Windows 95 将系统及应用的配置数据存储在登记簿 Registry 上。万一 Windows 95 启动失败,可以用 Windows 95 启动盘重新启动计算机,将上次存储下来的 Registry 备份拷贝到当前的 Registry 上。这可以恢复上次成功的设置项。

3) 字处理 Word 的安全措施

Microsoft Word 是 Microsoft 公司开发的文字处理软件。可以通过两种方法保护一个 Microsoft Word 文件:

(1) 读访问: 阻止其他没有口令的用户读文件;

(2) 写访问: 允许他人打开和读文件信息,但不能修改。

文件口令可以通过文件(File)菜单中的 Save as 界面中的选择项 Options 提供的功能建立或删除。

Microsoft Word 还提供了其他机会保护文件。例如,可以使用工具 Tools 菜单中的 Protect Document命令保护文件各部分。Microsoft Word 7 .0 的用户可以在 Help Answer Wizard 中找到 security 和 password,并在它的提示下进行操作。Microsoft Word 6 .0 的用户可以在 Help 中查找 password。

上述文件保护措施不能阻止他人用搜索器 Mac Finder 或文件管理器 Windows File Manager 删除文件。

4) 电子表格 Excel 的安全措施

Microsoft Excel 是 Microsoft 公司开发的电子表格处理软件。与 Microsoft Word 一样,可以通过两种方法来保护一个 Excel 文件:

(1) 读访问: 阻止其他没有口令的用户读电子表格信息;

(2) 写访问: 允许他人打开和读电子表格信息,但不能修改。

文件口令可以通过文件(File)菜单中的 Save as 界面中的选择项 Options 提供的功能建立或删除。如果在 Write Reservation Password 域输入了一条口令,其他人在没有正确

口令时只能读,但不能修改或写入数据。

上述文件保护措施不能阻止他人用搜索器 Mac Finder 或文件管理器 Windows File Manager 删除文件。

5.2 数据库的安全与保密

本节介绍数据库的安全管理策略、存取控制策略及数据库的加密方法并介绍了数据库安全保密实例。

1. 安全数据库的方法

数据库系统通常由数据库和数据库管理系统两部分组成。前者由记录构成,每个记录有一组相联系的属性数据;后者是一个软件,它帮助用户建立、使用和管理数据库。

安全数据库的基本要求可归纳为:数据库的完整性(物理数据库的完整性、逻辑数据库的完整性和数据库中属性数据的完整性)、数据库的保密性(用户身份识别、信息保密、访问控制和可审计性)、数据库的可用性及有用性(用户界面友好,在授权范围内用户可以简便的访问数据)。通常,可从安全管理和存取控制两方面保证数据库的合法使用。

1) 安全管理策略

实现安全管理的方法一般有集中控制和分散控制两种方式。对于前者,是由单个授权者控制系统的整个安全维护的各个方面;对于后者,则是采用不同的管理程序控制数据库的各个部分。在某些环节中,这种方式更为有效和方便。

2) 存取控制策略

存取控制策略包括以下几点。

(1) 最小特权策略　该策略是数据库安全保护的最主要的策略,适合于任何系统的安全保护。该策略使用户只了解自己工作所需的信息,对于其他信息都加以屏蔽和保护,使信息泄漏的可能性最小,从而使数据库完整性受到损害的程度也最小。最小策略也称为"知所必需"的策略。

(2) 最大共享策略　它是一个基本的可供选择的策略,使数据信息得到最大程度的共享。这要求在保密控制的条件下得到最大的共享,不是任何人都可以随意存取数据库中的信息。

(3) 开放与封闭系统　存取控制的一种策略问题就是究竟是建立一个开放系统,还是建立一个封闭系统。所谓开放系统,只当明确地禁止时,才不能对该系统进行存取操作。封闭系统正好相反,仅当明确地授权后才能对该系统进行存取操作。从安全保密的角度来讲封闭系统具有比较安全的特点,且符合"最小特权"原则。

(4) 按名存取控制　这种控制策略是最小特权策略的扩展,能达到细化求精的目的。它要求数据库管理系统提供允许的最小的"颗粒",以达到最小特权的存取控制。

(5) 取决于上下文的存取控制策略　该策略是指存取控制项的组合。一方面它限制了组合在一起的存取域;另一方面有时又要求某些属性组合在一起存取。

(6) 取决于历史的存取控制策略　该策略控制存取过程,在存取时,不仅要考虑当前

的请求,还要考虑用户过去的存取历史。因为有些数据本身不会泄漏秘密,但是如果和以前历史上查询得到的数据联系起来,就可能泄漏机密。可以根据用户过去已经执行的存取来拒绝他现在的存取请求。这一策略对统计数据库安全非常重要。

2. 数据库的加密方法

上面介绍的安全策略为数据库的安全性提供了一定的保障,但是有经验的攻击者可能会借助某种手段避开应用程序而直接进入系统访问数据。为了防止这样的信息泄漏,可以采取对数据库进行加密的手段,将其变成密文。对数据库的加密可从三方面进行。

1) 库内加密

库内加密的一种方法是数据元素加密:将一条记录的某个属性值作为一个文件进行加密。例如,在我们设计的"通用智能题库系统"中,设计了试题数据库及答案数据库,对于试题流水号、答案流水号属性值不必加密,而对试题内容、答案内容需要加密,以确保题库安全(详见本节第 5 部分内容)。另一种方法是记录加密:将一个记录作为一个文件而加密。

2) 整个数据库加密

将整个数据库包括数据库结构及数据库内容当作一个文件进行加密。在这种加密方式中加密、解密及密钥管理比较简单。但也存在一定的缺点,例如,有时仅访问数据库中某一记录的某一属性值,也要解密整个数据库。

3) 硬件加密

设计硬件来实现数据库的加密。

3. 数据库的恢复

尽管采用了一些保护措施来防止数据库完整性和安全性被破坏,但有时一些破坏是不可避免的,数据库恢复技术是一种可采取的补救措施。数据库恢复具体可采用下面三种方法:

· 利用操作系统提供的功能,将被错误删除或修改的数据恢复,或将送到回收站的数据恢复。

· 定期地将整个数据库复制到软盘上保存起来,或刻录到光盘上保存起来,当数据库遭到破坏后,就可以利用备份将数据恢复。

· 利用各数据库之间的关系,用未遭到破坏的数据库恢复已遭到破坏的数据库。

4. Microsoft Access 数据库的安全保护

Microsoft Access 是美国微软公司的数据库产品,内嵌了大量的保护特性,且大多数保护特性是针对计算机网络的。Microsoft Access 提供了用户标识、确认、加密、授权及审计等许多控制来实现对数据库的安全保护,保护措施存储在 Security Wizard 中。Microsoft Access 使用 RSA 算法给数据库加密,以便只有授权用户才能使用它。Microsoft Access 提供的数据库加密可以按下列步骤来操作:

步骤 1　从 Microsoft Access 中获得加密向导 Security Wizard;

步骤 2　运行 Security Wizard;

步骤 3　选择文件 File 菜单中的 Encrypt/ Decrypt Database 项和带有加密选项的数据库文件实施数据库加密。

5. 数据库安全保密实例——通用智能题库安全保密的实现

信息系统的开发与设计离不开数据库的设计与维护,因此对数据库的安全保密问题尤为重要。下面以通用智能题库系统国家教委八·五重点科研课题安全保密的实现为例说明数据库加密的实用技术。

计算机密码学的发展和应用,为题库的安全保密工作提供了有力的支持。将计算机密码学的基本原理和方法应用于题库建设,可以有效地防止泄密事件的发生,保障题库的安全。智能题库的安全保密子系统正是基于这种思想开发的。

针对一个通用智能题库系统的安全与保密问题,提出了一个适用于题库系统的安全保密系统,并证明了该系统的安全性。该系统具有身份识别、访问权限控制、信息加密、解密以及密钥管理的功能。还提出了一个新的单向加密算法和一种针对数据库的访问控制策略。通用智能题库的保密系统的结构如图 5.10 所示。下面就其中的几个主要部分进行介绍。

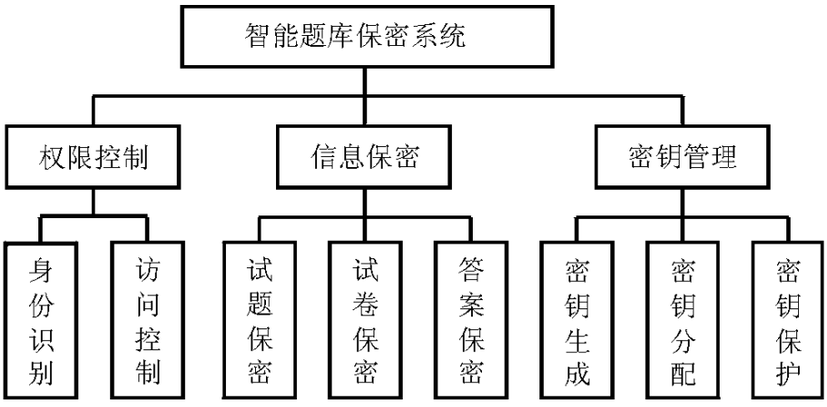


图 5.10　通用智能题库保密子系统的结构

1) 通用智能题库中安全保密机制

在通用智能题库中安全保密机制实施方案如图 5.11 所示。

2) 访问权限控制

访问权限控制分为身份识别和访问控制两部分。

(1) 身份识别

用户进入系统前首先要识别用户的身份,这个过程类似于操作系统的注册过程。

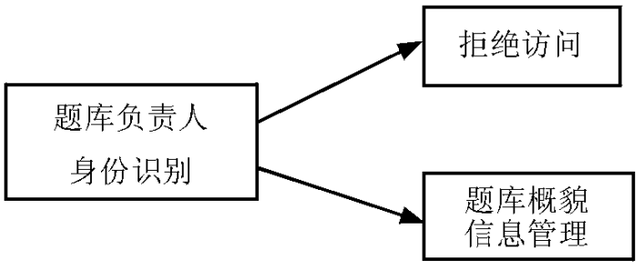
这里,利用指数密码构造了一个单向函数,用这个单向函数来加密用户的注册口令,将密文连同用户的标识号一起存入系统的口令文件中。它的保密强度是基于有限域中的离散对数算法的复杂性。

基于用户标识和口令 P 的单向识别注册协议描述如下:

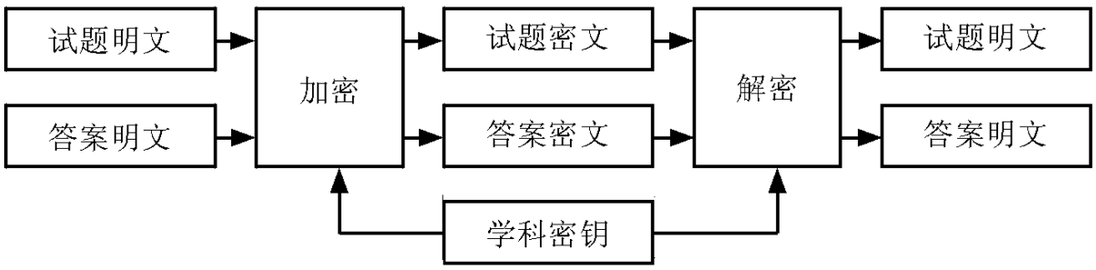
步骤 1　请用户输入标识符 ID 和口令 P;

步骤 2　查找口令文件,找到与 ID 值对应的密文 Y;

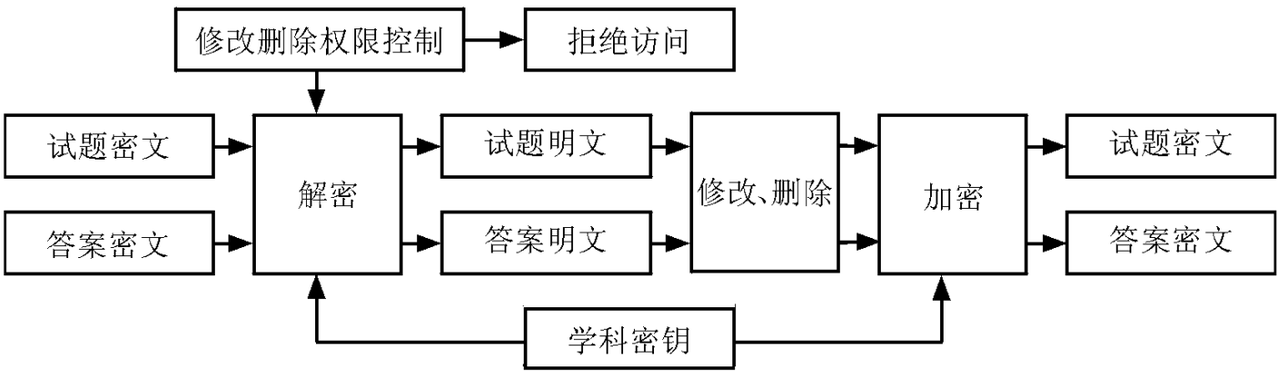
步骤 3　将口令 P 变换成整数 P';



(a) 身份识别方案



(b) 试题及答案信息保密方案



(c) 修改删除权限控制及信息保密方案

图 5.11　安全保密机制实施方案

步骤 4　加密 P': $X = f(p') \equiv m_0^{p'} \bmod n$;

步骤 5　判断是否有 $X = Y$? **若是,则注册成功,否则注册失败。**

(2) 访问控制

针对题库的访问控制问题,本系统提出了一个访问控制策略:

· 在录入每道题的参数、试题文字及答案文字之后,对题库中每道试题的试题部分(文字和图形)与答案部分(文字和图形)分别用相同的密钥采用 DES 算法进行加密,形成密文。在数据库中存储的是密文信息,攻击者即使得到数据库,由于不知道密钥,也就无法知道试题及答案信息本身具体是什么内容。

· 每门课程使用一个密钥加密,密钥存放在软盘上由专人妥善保管。

· 在查询试题及答案信息或自动排版出试卷清样时,首先要求用户插入相应课程的密钥盘,用该密钥对密文信息进行解密,变成试题或答案的明文,进而显示试题或答案信息的明文。

· 当用户要求修改试题或答案内容时,除了要有该课程的密钥之外,还要求用户拥有修改权限,这是通过给有相应权限的人员分配一个修改密钥实现的,该密钥也存放在软盘上,由相应人员负责保管,验证密钥采用上述的单向注册协议。如果修改、删除身份识别

正确,那么可以进行相应的修改、删除操作。如果身份不对,则拒绝操作。

· 对于题库概貌信息的管理,也有对该项操作的权限控制及身份识别。如果身份不对,则拒绝访问;如果身份正确,那么可以进入题库概貌信息管理子系统,进行相应的操作,例如,浏览题库概貌信息等。

3) 信息保密

对题库中试题和答案的加密、解密是利用数据加密标准算法 DES 来实现的。DES 算法被美国国家标准局宣布为商用数据加密标准算法,保密强度较高,实现速度也比较快。

4) 数据库中数据元素的加密

智能题库系统中有许多数据库,这里以试题文字数据库和答案文字数据库为例,讨论数据库中数据元素的加密、解密处理过程算法。表 5 .2 和表 5 .3 给出了这两个数据库的结构。

表 5 2 试题文字数据库结构

序号	字段名称	字段代码	类型	宽度
1	流水号	LSH	Numeric	6
2	试题文字	STWZ	Memo	10

表 5 3 答案文字数据库结构

序号	字段名称	字段代码	类型	宽度
1	流水号	LSH	Numeric	6
2	答案文字	DAWZ	Memo	10

对于数据库中试题文字、答案文字字段的属性值进行加密、解密,可以由下列过程算法实现。

试题及答案信息加密处理过程算法描述如下:

步骤 1 录入或修改某门课程的一道试题及答案信息,分别形成文件,如 st .doc 和 da .doc

步骤 2 实施加密,原理如下:

stcipher = EN - DES(st .doc, Key)

dacipher = EN - DES(da .doc, Key)

其中加密、解密采用 DES 标准数据加密算法, EN - DES 为加密变换, Key 是课程密钥, stcipher 和 dacipher 分别为试题和答案信息加密后的密文。

具体操作可以使用下列语句实现:

run/ n c: \ tiku \ screen \ en - des .exe c: \ tiku \ screen \ st .doc c: \ tiku \ screen \

stcipher .doc

run/ n c: \ tiku \ screen \ en - des .exe c: \ tiku \ screen \ da .doc c: \ tiku \ screen \

dacipher .doc

其中 en_des .exe 是一个实现 DES 算法的加密器,c: \ tiku \ screen 是存储题库及加密器和解密器的路径。

步骤 3　把密文 stcipher 及密文 dacipher 作为数据库的一个 Memo(备注)字段而存储。具体操作可以使用下列语句实现:

append Memo stwz from stcipher .doc

append Memo dawz from dacipher .doc

试题及答案信息解密处理过程算法描述如下:

步骤 1　从试题信息数据库 st .dbf 及答案信息数据库 da .dbf 中提取 stwz 及 dawz 字段内容,形成密文文件。

具体操作可以使用下列语句实现:

copy Memo stwz to stcipher .doc

copy Memo dawz to dacipher .doc

步骤 2　实施解密,原理如下:

st = DI_DES(stcipher .doc, Key)

da = DI_DES(dacipher .doc, Key)

其中 DI_DES 是解密变换,采用与加密相同的密钥,对试题、答案密文信息进行解密,形成明文信息,供查询及自动排版出清样等使用。

具体操作可以使用下列语句实现:

run/ n c: \ tiku \ screen \ di_des .exe c: \ tiku \ screen \ stcipher .doc c: \ tiku \ screen \ st .doc

run/ n c: \ tiku \ screen \ di_des .exe c: \ tiku \ screen \ dacipher .doc c: \ tiku \ screen \ da .doc

其中 di_des .exe 是一个实现 DES 算法的解密器。

5) 密钥管理

(1) 密钥的产生

· 56 bit DES 密钥的产生:利用随机数产生器随机产生 56 bit 0 - 1 序列。

· 各访问控制密钥的产生:利用随机数产生器随机产生 40 位十进制整数作为密钥。随机选择 40 位十进制大素数作为模。

(2) 密钥的分配

身份验证、试题修改和题库概貌管理的访问控制密钥在系统初始化时产生并分配给用户,各课程密钥在试题录入时产生并分配给用户。

6) 系统安全性分析

(1) 单向函数体制的安全性

单向函数体制的保密强度是基于有限域中离散对数算法的复杂性。这里,素数模 n 取 40 位十进制整数。由于 n 是保密的,所以猜测 n 的计算复杂度为 $O(10^{38})$,猜测口令的计算复杂度为 $O(10^{29})$,因而用穷举法来破译是不可能的。

(2) DES 算法的安全性

DES 算法已经成为美国国家标准算法,从它提出那天起,就一直有人在试图系统地破译它,但直到今天也没有人能够破译,其保密强度是足够高的。

5.3 数 字 签 名

本节首先介绍数字签名及其特点,然后重点介绍 DSA 数字签名算法,接着给出一个使用 DSA 生成、验证签名的例子。

1. 数字签名及其特点

在信息化社会,签名盖章和识别签名是一个重要环节。例如,银行业务、电子资金传送、股票、证券交易、合同、协议的签字等,都需要签名。在计算机广泛应用的时代,应用密码学的方法实现数字签名,具有重要的理论意义,更具有重要的实际意义。

假设发送方 sender 发送了一个签了名的信息 message 给收方 receiver,那么 sender 的数字签名必须满足下述条件:

1) receiver 能够证实 sender 对信息 message 的签名;

2) 任何人包括 receiver 在内,都不能伪造 sender 对 message 的签名;

3) 假设 sender 否认对信息 message 的签名,可以通过仲裁解决 sender 和 receiver 之间的争议。

假定 sender 向 receiver 发送一则消息 message,采用公开钥密码系统的签名过程描述如下:

1) sender 计算: c = Da(message)对 message 签名。Da 是解密变换,所使用的秘密钥为 sender 所私有,任何人,包括 receiver 在内,由于不知道 Sender 的秘密钥,所以不能伪造 sender 的签名。

2) receiver 通过检查 Ea(c)是否恢复 message 来验证 sender 的签名,Ea 是 Da 的逆变换,Ea 变换中所使用的密钥是 Sender 的公开钥。

3) 如果 sender 和 receiver 之间发生争议,仲裁者可以用 2)中的方法鉴定 sender 的签名。

例: sender 表示一个银行电子支付系统的客户,receiver 代表 sender 所在的银行。当 receiver 收到 sender 要求取款 1000 万日元的信息后,必须鉴定这个信息确实是由 sender 签名发出的。如果以后 sender 否认这一笔取款,receiver 能够向仲裁者证实,这个取款单确实是由 sender 签署的。如果采用公开钥密码系统签名,秘密钥仅由 sender 所拥有,公开钥大家都知道,sender 无法抵赖否认自己的签名。别人由于不知道秘密钥也无法冒名顶替 sender 的签名。在此例中,由于银行和储户之间的业务信息往来应秘密进行,所以这个数字签名系统除要求生成签名与验证外,还要求加密。然而,并非所有的数字签名系统都要求保密性。

2. 数字签名算法 DSA

DSA(digital signature algorithm)算法是美国国家标准与技术学会(National Institute

of Standards and Technology, NIST) 于 1994 年公布的一个数字签名标准。DSA 是一个公开钥数字签名算法,用于检验数据的完整性和一致性,第三方可以使用它来确认数字签名的合法性。

1) DSA 算法描述

DSA 是 Schnorr 和 ELGamal 数字签名算法的变体,DSA 算法使用的参数含义如下:

p: L 比特长的素数,其中 L 范围是从 512 比特到 1024 比特,并且要求是 64 的整数倍(在原始标准中 p 的尺寸固定在 512 比特,后来 p 的尺寸由 $NIST$ 做了改变)。

q: 160 比特的数,并且要求是 $p-1$ 的素数因子。

g: $g \equiv h^{(p-1)/q} \bmod p$,其中 h 是小于 $p-1$ 的任意数,并且 $h^{(p-1)/q} \bmod p > 1$

x: 小于 q 的数

y: $y \equiv g^{x} \bmod p$

前三个参数 p, q 和 g 是公开的,可以由一组网络用户共享。秘密密钥为 x,公开密钥为 y。DSA 算法还利用了单向哈希函数 $H(m)$,该标准还规定了安全哈希算法。

对信息 m 进行签名的过程可以描述如下:

(1) 发送方 sender 生成一个随机数 k,并且 $k < q$

(2) 发送方 sender 生成签名:

$$r = (g^{k} \bmod p) \bmod q$$

$$s = (k^{-1}(H(m) + xr)) \bmod q$$

参数 r 和 s 是 sender 的签名,sender 把这些信息发送给接收方 receiver。

(3) receiver 通过计算验证这些签名:

$$w = s^{-1} \bmod q$$

$$u_1 = (H(m) * w) \bmod q$$

$$u_2 = (r * w) \bmod q$$

$$v = ((g^{u_1} * y^{u_2}) \bmod p) \bmod q$$

如果 $v = r$,那么签名被证实。

2) 使用预先计算来加快速度

DSA 的实现经常是通过预先计算来加快速度的。注意到 r 的值不取决于信息。可以生成一串 k 值,然后再计算对应的 r 值,也可以再计算 k 对应的 k^{-1}。于是,当对消息签名时,可以用给定的 r 和 k^{-1} 进行计算。

表 5.4 给出了 DSA 样本软件速度。

表 5.4 对于不同的模长度带有 160 比特指数的 DSA 速度

模长度	签名	验证
512 比特	0.20 秒	0.35 秒
768 比特	0.43 秒	0.80 秒
1024 比特	0.57 秒	1.27 秒

如果 p 取 512 比特,DSA 算法对长期安全强度是不够的,如果 p 取 1024 比特,DSA 算法对长期安全强度是足够的。

3) DSA 素数生成

密码学家 Lenstra 和 Haber 指出 DSA 的一些模是容易攻破的。如果一些网络用户使用这样的模,那么他的签名就会被伪造。但是这些模很容易被检测,而且数目很少,以至于选择这样的模的概率很小。事实上比使用概率素数生成模式生成一个合数的机会还小。

NIST 学会推荐生成两个素数 p 和 q 的一个特别的方法,其中 q 能整除 $p-1$, p 长度为 L 比特, L 范围是 512 比特到 1024 比特之间,并且是 64 的整数倍。素数 q 长度是 160 比特。令 $L-1=160n+b$,其中 L 是 p 的长度, n 和 b 是两个数, $b<160$。DSA 素数生成步骤如下:

(1) 选择至少有 160 比特的任意序列,称之为 S。令 g 是 S 的比特长度;

(2) 计算 $U=SHA(S)\quad SHA((S+1) \bmod 2^g)$,其中 SHA 是安全哈希算法;

(3) 设置 U 的最高有效位及最低有效位均为 1,形成 q;

(4) 检查 q 是否是素数;

(5) 如果 q 不是素数,则转回步骤(1);

(6) 令 $C=0$ 且 $N=2$;

(7) 对应 $k=0,1,2,\cdots,n$,

令 $V_k=SHA((S+N+k) \bmod 2^g)$;

(8) 令 W 是一个整数

$$W=V_0+2^{160}V_1+\cdots+2^{160(n-1)}V_{n-1}+2^{160n}(V_n \bmod 2^b)$$

并且令 $X=W+2^{L-1}$;

(9) 令 $p=X-((X \bmod 2q)-1)$,要求 $p \bmod 2q=1$;

(10) 如果 $p<2^{L-1}$则转到步骤(13);

(11) 检查 p 是否是素数;

(12) 如果 p 是素数,则转到步骤(15);

(13) 令 $C=C+1$ 且 $N=N+n+1$;

(14) 如果 $C=4096$, 则转到步骤(1),否则转到步骤(7);

(15) 存储 S 和 C 的值,用于生成 p 和 q。

上式中,变量 S 称作“种子”, C 称作“计数器”, N 称作“偏移量”。

4) 使用 DSA 的 RSA 加密

假设 DSA 算法由一个单独的函数调用实现:

DSAsign(p, q, g, k, x, h, r, s)

给函数提供参数 p, q, g, k, x, h,函数返回签名 r 和 s。

使用 RSA 加密是比较容易的,设模为 n,消息为 m,公开密钥为 e,调用

DSAsign(n, n, m, e, 0, 0, r, s)

那么返回的 r 值就是密文,令 $c=r$。

相应的,用 RSA 实现解密也是比较容易的,如果 c 为密文, d 为秘密密钥,调用

DSAsign(n, n, c, d , 0, 0, r, s)

那么返回的 r 值就是明文。

5) DSA 变体

为了突出不同方面的实现特性,下面介绍几种 DSA 变体。

第一种变体使签名更容易实现,它是通过使用户不去计算 k^{-1} 来实现的,DSA 变体的所有参数与 DSA 中的参数相同。为了对信息 m 签名,发送方生成两个随机数 k 和 d,且 $k<q$, $d<q$。签名:

$$r \equiv (g^k \bmod p) \bmod q$$
$$s \equiv ((H(m) + xr) * d) \bmod q$$
$$t \equiv (kd) \bmod q$$

receiver 通过计算验证这些签名:

$$w \equiv (t' s) \bmod q$$
$$u_1 \equiv (H(m) * w) \bmod q$$
$$u_2 \equiv (rw) \bmod q$$

如果, $r \equiv ((g^{u_1} * y^{u_2}) \bmod p) \bmod q$ 那么签名被证实。

第二种 DSA 变体使验证者计算更容易。DSA 变体的所有参数与 DSA 中的参数相同。为了对信息 m 签名,发送者生成一个随机数 k,且 $k<q$, 签名:

$$r \equiv (g^k \bmod p) \bmod q$$
$$s \equiv k * (H(m) + xr)^{-1} \bmod q$$

receiver 通过计算验证这些签名:

$$u_1 \equiv (H(m) * s) \bmod q$$
$$u_2 \equiv (sr) \bmod q$$

如果 $r = ((g^{u_1} * y^{u_2}) \bmod p) \bmod q$,那么签名被证实。

第三种 DSA 变体允许成批地验证。接收方成批地验证签名。如果都是有效的,则结束;如果某一个无效,接收方还得去找到这个签名。遗憾的是,这种方法不安全。签名者和验证者很容易生成一组伪造的签名来满足验证的标准。

第四种变体是 DSA 素数生成的变体,即

(1) 选择至少有 160 比特的任意序列,称之为 S。令 g 是 S 的比特长度;

(2) 计算 $U = SHA(S) \quad SHA((S+1) \bmod 2^g)$,其中 SHA 是安全哈希算法;

(3) 设置 U 的最高有效位及最低有效位均为 1,形成 q;

(4) 检查 q 是否是素数;

(5) 令 p 是 q, S, C 和 $SHA(S)$的并置链接, C 被设置成 32 比特,并且全为 0;

(6) $p = p - (p \bmod q) + 1$;

(7) $p = p + q$;

(8) 如果 p 中的 C 是 0X7fffffff,那么转到步骤(1);

(9) 检查 p 是否是素数;

(10) 如果 p 是合数,则转到步骤(7)。

关于这个变体,可以不必存储用于生成 p 和 q 的 S 和 C 的值,它们与 p 嵌在一起。对于应用,不需要大量的存储器,这一点很可贵。但这种模式是否降低了 DSA 的安全性,

目前还不十分清楚。

3. 使用 DSA 生成、验证签名的例子

这里介绍使用 Java 语言实现使用 DSA 生成、验证签名的一个例子。

1) Java 简介

谈论 Internet 和 Intranet 或网络设备而不谈论 Java 是不可能的。似乎一夜之间 Java 成为建立网络化应用的标准平台,Java 是一种通用的、并发的、基于类的、面向对象的程序设计语言,它被设计得足够简单以便许多程序员很容易地使用它。Java 是一种生产性语言而不是研究性语言,Java 使用垃圾收集器技术自动地进行内存管理,避免不安全因素。Java 程序设计语言和历史上任何其他程序设计语言相比被采用得最快,并且 Java 技术被大部分公司认作一个关键技术来支持平台独立存取广泛的协作资源。

无疑 Java 在软件开发者中越来越受欢迎。它对 Internet 和 Intranet 应用的开发和使用起到了革新的作用。它提供了一种新的、令人兴奋的面向对象的网络计算模型。这种模型易于理解,使用这种模型很容易地进行程序设计。Java 取得了巨大的成功的另一个原因就是:用 Java 写一次程序,测试一次,然后可以到处运行(write a program once, test it once, and then run it anywhere),而不需考虑所用的硬件和操作系统。这可以实现在网络上软件、信息的共享和再利用。这个事实也蕴含一个潜在的优势:软件开发时间和成本的巨大节省。

Java 开发工具 JDK1 .1 和 Java Beans 开发工具(Beans development kit, BDK)1 .0 版的发表标志 Java 平台达到了一个新的成熟水平。在这次新的发表中 Java 虚拟机(Java virtual machine,JVM)被重写,使 Java 程序运行速度比 JDK 1 .0 .2 中发表的 JVM 运行速度快两倍或三倍。在 JDK1 .1 中有很多新的特征,所有的核心 API(application program interface,应用程序接口)在所有的平台和操作系统上都可以得到并且使用。这些新的特征包括 APIS 国际化、安全和签名的 Applets、抽象窗口工具(abstract windows toolkit, AWT)加强、Java 豆(JavaBeans™)、Java 档案文件格式、网络功能加强、输入/ 输出功能加强、数学软件包(Big Integer, BigDecimal 类)、远程方法调用、对象有序化、映象、Java 数据库连接(JDBC™)、内部类等。

Java .math .BigInteger 提供了任意精度的大数的模代数运算,如最大公约数计算 gcd(BigInteger),素数生成测试 BigInteger(int bitLength, int certainty, Random rnd),余数计算,指数运算 modPow(BigInteger),乘法逆元素计算 modInverse(BigInteger)。使用这些方法进行程序设计是很方便的,这就是使用 Java 的优点之一。如果使用其他的程序设计语言,这些都要自己编程。

Java 安全 API 提供加密、信息融合、密钥管理、认证、存取控制和数字签名功能,允许开发者进行低层和高层的安全应用。

2) 使用 DSA 生成、验证签名的例子

使用 Java 安全 API(应用程序接口)为数据生成一个签名,并验证签名是真实的。

步骤 1 准备初始化程序结构

数字签名的方法是包括在 java .security 的软件包中的,所以要输入来自于这个软件

包的一切。还需要输入 java .io 软件包,因为它包括了输入文件数据所需要的方法。

```
import java .io .* ;
    import java security .* ;
```

步骤 2　生成公开钥和秘密钥

为能生成签名,必须做的第一件事就是生成密钥对: 秘密钥和对应的公开钥。密钥由随机数生成器生成。秘密钥用于生成数字签名,对应的公开钥用于验证签名。密钥对是通过使用 KeyPairGenerator 类来生成的。

在这个例子中,为“ DSA ”算法生成一个公开的和秘密的密钥对。生成的密钥具有 1024bit 长度。

生成一个密钥对步骤如下:

(1) 创建一个密钥对生成器

为 DSA 签名算法生成密钥的第一步是获得一个密钥对对象。

```
KeyPairGenerator KeyGen = KeyPairGenerator .getInstance(“ DSA ”)
```

(2) 初始化密钥对生成器

所有的密钥对生成器都共享“ 强度 ”(strength)和“ 随机数源 ”(source of randomness)的概念。KeyPairGenerator 类中的方法 initialize 有两个类型变量。对 DSA 密钥对生成器强度 Strength 设置为 1024,随机数源必须是 Java 中的 Secure Random 类的一个实例。简单地说,可使用 Secure Random 的空的构造函数,这将自动地生成一个随机数生成器所需要的一个“ 种子 ”(seed)值。

```
KeyGen .initialize(1024 ,new SecureRandom() );
```

(3) 生成密钥对,并在 KeyPair 类的实例中存储密钥

```
KeyPair pair = KeyGen .GenerateKeyPair( );
```

步骤 3　对数据进行签名

在生成秘密钥和公开钥之后,准备对数据进行签名。在这个实例中,将对文件中包括的数据进行签名,可以从命令行得到文件名。使用签名 Signature 类的例子生成数字签名。

其签名步骤描述如下:

(1) 得到一个签名对象(object)

通过使用下面的语句可以得到一个使用 DSA 算法生成和验证签名的签名对象。

```
Signature dsa = Signature .getInstance(“ SHA/ DSA ”);
```

一般地,说明签名算法名称时,也应该说明这个签名算法所使用的信息融合算法名称。DSA 算法被定义使用 SHA - 1 信息融合算法。“ SHA ”经常被参考成 SHA - 1 算法。

(2) 对签名对象进行初始化

在签名对象能用于签名(或验证)以前,必须将它进行初始化。签名的初始化方法需要秘密钥,可以从前一步生成的密钥对中抽取和使用秘密钥。

```
PrivateKey Priv = Priv .getPrivate( );
dsa .initSign(Priv);
```

(3) 给签名对象提供要签名的数据

使用来自于文件的数据,对文件中的数据一次读一个 byte。文件名是在第一个命令行变量中说明的。通过调用 Update 方法把数据提供给签名对象。

```
FileInputStream fis = new FileInputStream(args[0]);
byte b;
while(fis .available( ) ! = 0)
{
    b = (byte)fis .read( );
    dsa .update(b);
}
fis .close( );
```

(4) 生成签名

一旦把所有的数据提供给签名对象,就可以对数据生成数字签名:

```
byte[ ] sig = dsa .sign( );
```

步骤 4　验证签名

如果已经具有了经过数字签名的数据,通过使用 Java 中的 Security 就可以验证这个签名的真实性。验证签名需要以下三项内容:数据、签名和对应签名所使用的秘密钥的公开钥。

使用签名 Signature 类的一个实例:

```
Signature dsa = Signature .getInstance(“ DSA ”);
```

(1) 为了验证,首先对签名对象进行初始化

签名对象在签名时进行过初始化。现在,为了验证,必须将它进行初始化。用于验证的初始化方法需要一个公开钥,可以从步骤 2 产生的密钥对中抽取和使用公开钥。

```
PublicKey pub = pair .getPublic( );
dsa .initVerify(pub);
```

(2) 给签名对象提供要验证的数据

必须给签名对象提供要验证的数据。这个数据包括在文件中,文件名应是在第一个命令行变量中说明的。

正像签名时所做的,每次从文件中读一个 byte。通过调用 Update 方法,把数据提供给签名对象。像步骤 3 一样,使用 FileInputStream 变量 fis 及 byte 变量 b。

```
FileInputStream fis = new FileInputStream(args[0]);
while(fis .available( ) ! = 0)
{
    b = (byte)fis .read( );
    dsa .update(b);
}
```

```
fis .close( ) ;
```

(3) 验证签名

一旦给出签了名的对象,就可以证实这个签名的正确性。

```
boolean verifies = dsa .verify(sig) ;
System .out .println(" Signature verifies: "+ verifies) ;
```

一个完整的使用 DSA 方法进行数字签名的 Java 程序例子如下:

```
import java .io .* ;
import java security .* ;

class testsig{
    public static void main(string[ ] args){
        / * 生成和验证 DSA 签名的程序例子 */
        try{
            / * 生成密钥对 * /
                KeyPairGenerator KeyGen = KeyPairGenerator .getInstance(" DSA ") ;
                KeyGen .initialize(1024, new SecureRandom( )) ;
                KeyPair pair = KeyGen .generateKeyPair( ) ;
                / * 为了签名和验证,创立一个签名对象 * /
                Signature dsa = Signature .getInstance(" SHA/ DSA ") ;
                / * 为了签名和验证 ,对签名对象进行初始化 */
                PrivateKey priv = pair .getPrivate( ) ;
                dsa .initSign(priv) ;
                / * 给签名对象提供要签名的数据 * /
                FileInputStream fis = new FileInputStream(args[0]) ;
                byte b;
                while (fis .available( ) ! = 0)
                {
                    b = (byte)fis .read( ) ;
                    dsa .update(b) ;
                    } ;
                fis .close( ) ;
                / * 生成签名 */
                byte[ ] sig = dsa .sign( ) ;
                / * 验证签名 */
                / * 为了验证,对验证对象进行初始化 */
                PublicKey pub = pair .getPublic( ) ;
                dsa .initVerify(pub) ;
                / * 给验证对象提供要验证的数据 */
                FileInputStream fis = new FileInputStream(args[0]) ;
                while (fis .available( ) ! = 0)
                {
                    b = (byte)fis .read( ) ;
                    dsa .update(b) ;
                    } ;
                fis .close( ) ;
```

```
                    /* 验证签名 */
                    boolean verifies = dsa.verify(sig);
                    System.out.println("Signature verifies: " + verifies);
                    } catch (Exception e) {
                        System.err.println("caught exception" + e.toString());
                    }
                }
            }
```

4. 数字签名算法 GOST

这是一个俄罗斯数字签名标准,经常被称作 GOST R 34.10-94。这个算法非常类似于 DSA。它使用了下面的一些参数:

p 是一个素数,长度在 509 比特到 512 比特之间,长度或者在 1020 比特到 1024 比特之间。

q 是长度在 254 比特到 256 比特之间的数,并且是 $p-1$ 的素数因子。

a 是小于 $p-1$ 的数,并且 $a^q \bmod p \equiv 1$。

x 是小于 q 的数。

$y \equiv a^x \bmod p$。

算法中使用了单向哈希函数 $H(x)$。

前三个参数 p, q 和 a 是公开的,可以由一组网络用户共享。秘密密钥为 x,公开密钥为 y。

对信息 m 进行签名的过程可描述如下:

(1) 发送方 sender 生成一个随机数 k,并且 $k < q$

(2) 发送方 sender 生成签名:

$$r \equiv (a^k \bmod p) \bmod q$$

$$s \equiv (xr + k(H(m))) \bmod q$$

如果 $H(m) \bmod q = 0$ 那么令 $H(m) \bmod q = 1$。

如果 $r = 0$ 那么选择另一个 k,重新开始。

签名为两个数: $r \bmod 2^{256}$, $s \bmod 2^{256}$。sender 把这些信息送给接收方 receiver。

(3) receiver 通过计算验证这些签名:

$$v \equiv H(m)^{q-2} \bmod q$$

$$z_1 \equiv (sv) \bmod q$$

$$z_2 \equiv ((q - r) * v) \bmod q$$

$$u \equiv ((a^{z_1} * y^{z_2}) \bmod p) \bmod q$$

如果 $u = r$,那么签名被证实。

这种模式与 DSA 之间的区别在于: DSA 中 $S \equiv (k^{-1}(H(m) + xr)) \bmod q$,这导致一个不同的验证方程。令人好奇的是 q 取 256 比特。大部分西方密码学家似乎满意于 q 大约取 160 比特。也许这是俄罗斯人倾向于超安全的一个反映。这个标准从 1995 年以来一直使用,但它没有被用于特殊用途。

5.4 智 能 卡

1. 智能卡的发展

现代社会是信息化的社会,现代社会的经济是以知识和信息为基础的知识经济。社会的信息化和经济的知识化使信息的处理和存储的大容量、高保密、智能化及易携带性成为大势所趋,而日新月异的电子科技则使这一需求变为现实。1994 年法国人 RoLand Moreno 为了将一些个人信息存放在一个便于携带、保存的存储媒体上,提出了将一个集成电路芯片嵌入一个塑料基片上构成一张存储卡的想法,并按此方法做出了一张卡,这就是世界上第一张智能卡。从此开拓了智能卡的新时代。

智能卡是一种大小和形状与信誉卡相同的塑料卡片,卡片中嵌入了一个计算机芯片。早在 20 多年前,第一个关于智能卡的专利就已经存档,只是由于实用的局限性,近年才得以应用。目前许多国家使用智能卡办理电话业务。此外,智能信誉卡、智能兑付卡及其他功能的智能卡也相继使用。

智能卡的出现是超大规模集成电路和计算机技术高度发展的结果。将具有数据存储、处理、安全保密功能的集成电路芯片镶嵌到塑料卡片上便构成 IC 卡(integrated circuit card),因而称为集成电路卡。在亚洲地区通常称为“智能卡”。而在欧、美则通常称为 smart card 或 IC card。IC 卡除了存储数据外,还具有运算、处理和控制能力,被认为是世界上最小的个人计算机。

智能卡有不同的密码协议和编程算法,可以将智能卡看作是一个电子钱包,需要花钱时使用它。智能卡可以执行零知识验证协议,可以有自己的加密密钥,能对文件或计算机上未加锁的应用程序签名。

某些智能卡具有防篡改的功能,这样就能保障发卡机构的利益。因为对一个银行来说,它在发放智能卡时,肯定不希望用户能修改自己的卡而使自己账目上的存款更多。

2. 智能卡的种类和特点

1) 根据智能卡中所镶嵌的集成电路的不同可以将其划分为三类

(1) 存储器卡　卡中的集成电路为 E^2PROM。不需要密码操作,便能读写数据,一般用于保密程度不高的数据记录、采集及传递等领域。

(2) 逻辑加密卡　卡中的集成电路具有加密逻辑电路和 E^2PROM。必须经过一系列的密码核对才能对卡进行数据读写。这种卡的应用领域极其广泛,尤其在与钱财有关的领域得到广泛应用,例如,以此类卡为基础的“电子钱包”、“电子存折”等。

(3) CPU 卡　卡中的集成电路为一个单片微机系统,包括 CPU, RAM, ROM, E^2PROM, I/O 接口和片内操作系统 COS(chip operate system)。严格地讲,只有这类 CPU 卡才是真正的智能卡,并被誉为世界上最小的个人计算机。这类卡无论是其数据处理能力,数据存储容量、安全保密性能均远高于其他两类卡,代表着智能卡的发展方向。随着超大规模集成电路技术的发展,CPU 卡的成本在迅速下降,从而使其迅速成为智能卡的

主流。我国公安、税务、银行、医疗卫生等部门都颁布标准采用 CPU 卡。

智能卡中包含一个小计算机,通常是一个 8 位的微处理器,还有 250B 的 RAM 芯片,大约 6KB～8KB 芯片,几千字节的 EPROM 或 E^2PROM 芯片。将来的智能卡无疑会是功能更强大,但是智能卡本身的一些物理限制也使它的扩展比较困难。智能卡有自己的操作系统、程序和数据,智能卡的安全性在于它存放在用户的口袋里,别人是无法修改的。

2) 按卡与外界数据传送的形式不同可将其分为两类

(1) 接触卡　这种卡上的集成电路芯片有 8 个触点可与外界接触,卡与外界的数据传送通过与外界接触的芯片引脚进行。目前广泛使用的卡是这类接触型卡。

(2) 碰撞卡　这种卡是一种特殊的接触卡,不需要将卡插入卡座中,而仅仅需要将卡与卡座碰一下便可与外界交换数据。

3. 智能卡的应用前景

智能卡是一种镶嵌有大规模集成电路芯片的塑料卡片。由于智能卡的集成电路芯片具有数据处理、存储、通信、安全保密功能,被誉为世界上最小的个人计算机。虽然磁卡早于智能卡得到广泛应用,然而智能卡的应用正在迅速地增加,取代磁卡已成为不可阻挡的大势所趋。

由于智能卡具有数据存储量大,可靠性高,读写设备价格低,安全保密性能好,便于携带等突出优点,所以被广泛用于重要数据的存储媒体。例如,用智能卡存储个人身份信息,构成电子身份证、电子户口本、电子驾驶执照、电子出入证等电子证件。又例如,用智能卡存储金融货币信息,构成电子货币、电子存折、电子支票等电子货币凭证。总之,凡是涉及金钱支付或身份持证的地方都可以采用智能卡。

目前世界上使用智能卡普及的国家主要有法国、德国、西班牙、英国、日本、新加坡等经济发达国家,主要用于银行、税务、保险、医疗、电信、交通、身份证及各种预付费行业。我国政府正在推行“三金工程”之一的“金卡工程”,所谓“金卡工程”就是发展和普及智能卡应用从而实现货币电子化的工程。这是一项促进我国社会信息化和经济知识化的战略举措。显然,智能卡在我国的推广应用必然带来显著的社会效益和经济效益。

我国是一个人口大国,有着智能卡应用的巨大需求,因而推广智能卡的应用将对我国国民经济的发展和人民生活的现代化产生深远影响,并可产生巨大的经济效益。我国人力和知识资源丰富,劳动力成本低,因此我国智能卡制卡业特别是应用服务业在国际智能卡应用市场也具有竞争力。

4. 智能卡的安全问题

目前,与智能卡技术有关的主要安全问题是号码的窃取与篡改。智能卡技术的真正敌人是安全问题,最近,安全问题一直是媒体的焦点。

Mondex 是目前最安全的智能卡,卡的设计是防篡改的。智能卡技术使传统纸币和信用卡向前迈出一大步。但世界上没有一种技术能够完全避开安全问题。Mondex 智能卡的最佳应用是在 Internet 上进行小型购买,用户需要一个有效的付款工具,支付在 Internet 上类似 5 美分的购买行为。用户们害怕通过网络将他们的信用卡号码传送给类似

Time Warner 这样的厂商。但精明的黑客和安全专家并不担心。

当有人从 ESPN 和 NBA Web 用户处窃取了几千个信用卡号码时,许多人确实认为 Internet是一个危险的地方,不适合做生意。但这些号码是从服务器上窃取的,而不是在买方键入号码时被窃取的。安全专家们说,Internet 上的信息量非常大,单个信用卡号码不会被窃取。对于黑客来说,在 Internet 上搜索信用卡号码是毫无意义的,因为要得到包含信用卡数据的程序包非常困难。即使黑客能准确地找到此类程序包,但目前的多数金融事务处理都进行了安全插口层加密。黑客在计算机上花几小时时间解密数据以获得仅仅一个信用卡号码是毫无价值的。

要描述一个典型的黑客的攻击是不容易的。黑客入侵者的经验和技术水平有很大差异,各自的目的也不相同。某些黑客目的是要获取机密数据,有些黑客入侵者是故意给别人添麻烦,有些黑客入侵者仅仅是挑战。

当前全球网络发展的最新趋势是,一方面越来越多的基于不同目的有组织性的网络攻击开始出现,另一方面越来越多的"家贼"开始出现,而以前都是防范外部。据美国联邦调查局最近做的一次调查统计结果显示,目前一段时间 70% 的"黑客"行为是从企业内部发生的,只有 30% 是从外部攻进来的。许多用户以为装防火墙就万事大吉了,但防火墙主要是用来防范外部入侵的,有些用户采用了某些加密软件就以为安全了,其实这仅仅是部分安全,因此潜在的危害很大。

由于"黑客"这种行为是一种新出现的犯罪行为,在很多国家,包括在美国,都没有完整的立法,更为严重的是"黑客"的行为是国际化的,这种跨国的"黑客"行为就更没有相应的法律去评判。网络安全遭到"黑客"破坏后造成的后果包括金钱的损失、知识产权方面的损失等。目前美国正在进行这方面立法,在网络安全上网络拥有者应当承担的责任到哪一步。

安全电子事务 SET(secure electronic transactions)处理技术由 Visa 公司、MasterCard International公司、和其他主要信用卡公司以及硬件和软件厂商开发,旨在方便买方和卖方在 Internet 上安全地交换信用卡信息。虽然电子现金仍处于初级阶段,但金融机构需要一种安全标准。Herz 专家说:"安全威胁的感觉是存在的。如果我们要发展 Internet 商业的话,就必须解除人们的猜疑。"

5.5 EDI 系统的安全与保密

电子数据传输 EDI(electronic data interchange)技术,是一种先进的信息技术,它的产生、发展与应用,对信息安全保密提出了更高的要求。本节首先介绍 EDI 的基础知识,然后讨论 EDI 系统面临的威胁,最后提出 EDI 安全问题的相关对策和重要的法律保护条款。

1. EDI 的基本概念

EDI 系统能够将信息按照一定的协议,组织成标准格式,经过数据通信网络,在异地的计算机系统之间进行交换和自动处理。它从根本上废除了传统纸张邮递的往来方式,

充分体现了商业贸易与高科技的完美结合。

EDI 系统可以看作是信息系统的具体应用,它具有以下特点:

· 依靠通信网络,完成两个或多个计算机应用进程之间的通信;

· 通信中遵循一定的语法规则与国际标准;

· 信息是自动的传输交换,不需要人工干预,由应用程序对它自动响应,从而实现无纸贸易。

· 依靠信息管理的决策支持系统,完成综合信息的加工处理,实现事务处理自动化。

由此可见,通信网络是 EDI 系统的基础;计算机应用是 EDI 的条件;标准化法规是 EDI 的关键。

2. EDI 系统的功能

为了研究 EDI 系统安全问题,必须了解 EDI 系统的功能,完整的 EDI 系统通常包括以下功能模块:报文生成模块、信息处理模块、格式转换模块、通信模块和联系模块。各功能模块之间的关系如图 5.12 所示。

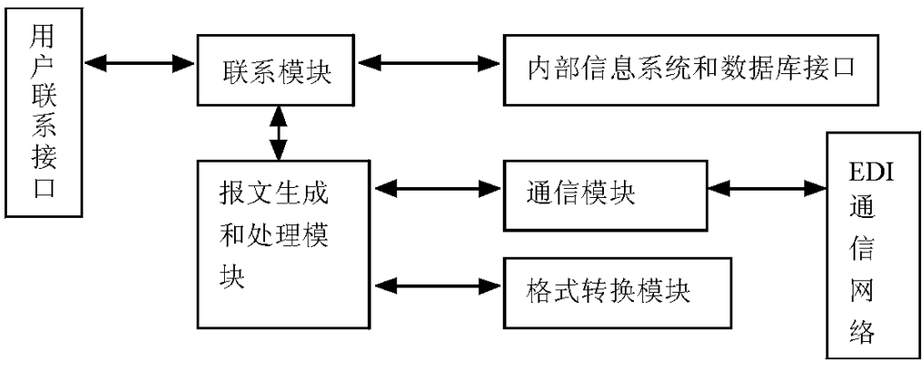


图 5.12　EDI 系统功能模块图

1) 报文生成模块

其功能是接受来自客户联系接口和其他信息系统,或数据库内部联系接口的命令和信息,然后按照 EDI 协议标准生成订单、发票、合同、许可证以及其他各种 EDI 报文,为信息的传输做好准备。

2) 信息处理模块

该功能是将接收到的其他 EDI 系统发来的报文,进行加工处理,按照不同的报文类型、不同的应用过程进行整理。例如:定单统计、发票汇总等。在处理过程中要与本部门的信息系统或数据库相结合,获取必要信息,以回复发方的 EDI 系统,并把有价值的信息送至本部门的信息系统和数据库。但是在信息处理过程中,若遇到意外情况,例如:交货时间变更、产品样式改变等方面问题,需要管理人员决策时,则将这类事件提交给用户联系接口处理。

3) 格式转换模块

该功能是将各种 EDI 报文,按照 EDI 结构化的要求,进行结构化处理,再遵循 EDI 语法规则进行压缩、重复、嵌套和代码转换,并附加上语法控制后,提交给通信模块,然后发送给其他 EDI 系统的用户;或者将其他 EDI 系统经通信模块接收到的结构化的 EDI 报

文,进行非法结构化的处理,以便使信息系统或数据库识别。格式转换应该具有相容性,满足不同国家不同地区的 EDI 标准。

4) 通信模块

它是 EDI 系统与通信网络之间的接口,完成呼叫、应答、自动转发、地址转换、差错校验、出错报警、认证审计、命名和寻址、合法性与完整性检查,发送报文等。由于通信网络的结构不同,对通信模块的要求也不同。

5) 联系模块

这部分包括用户联系模块和用户内部系统联系模块两个部分。用户联系模块是 EDI 系统与用户的联系接口,它为用户提供了友好的界面和人机会话环境。用户联系方式通常采用菜单驱动方式,使用户方便迅速的实现 EDI 的主要功能。另外联系模块可以完成用户所需的统计查询等工作,以便管理人员做出正确的决策。内部系统联系是 EDI 系统与本部门其他信息系统和数据库的接口,已经处理后的 EDI 报文,经过内部联系模块送往本部门的信息系统或数据库。一个部门的信息系统应用程度越高,内部系统联系模块也就越复杂。

一般来说,通信模块和格式转换模块对于所有的 EDI 系统应该是相同的,而报文生成模块、信息处理模块和联系模块因国家和地区的不同而有所差异,但是随着 EDI 标准化技术的不断发展,这些功能模块也将逐渐规范化。

3. EDI 系统的安全问题

随着 EDI 系统的广泛应用,彻底改变了纸张商贸的交易方式,于是传统贸易中的签字、盖章、笔迹等严密可靠的安全机制,对于 EDI 系统几乎完全不适用了。在 EDI 系统中,所有信息都是以数据文件的形式存储、传送,只要涂改,毫无痕迹。因此,如何解决好 EDI 系统的安全保密问题显得尤为重要,它可以保证 EDI 系统信息的处理、传送、存储和接收完整可靠,禁止非法人员的盗窃和篡改,从而促进 EDI 系统的普及与发展。下面让我们看一下 EDI 系统面临的威胁。

1) 非法冒充　这是 EDI 系统较为常见的破坏现象,非法用户伪装成合法用户对 EDI 系统进行访问,接收或发送消息,称为窃收。

2) 修改信息　EDI 信息在设备没有监视的情况下极易被篡改,非法者将信息的标识、内容、属性、接收者和发送者进行修改,造成混乱。

3) 重放信息　非法者将 EDI 信息窃取下来,修改之后再重放出来,达到破坏的目的。

4) 否认抵赖　用户拒绝承认已经发送的信息包括源否认、提交否认和接收否认。对于合同、契约、账单等贸易内容的否认,可能造成十分严重的损失。

5) 丢失信息　信息在 EDI 系统传输过程中丢失,有些信息可以重新补发,但有些信息可能无法重现,永久消失了。

6) 拒绝服务。它是指一个实体不能执行其功能或阻止其他实体执行其功能。拒绝服务表现为: 拒绝访问、拒绝通信、故意隐瞒信息,从而使局部系统失误和整个配合系统不一致,引起事故造成中断。

7) 统计分析　这是在盗窃信息的基础上进行的破坏活动,主要目的是对源信息进行

统计分析、加工整理,以求得到对其有价值的信息。

为了防止上述 EDI 威胁的出现,在 EDI 系统的设计过程中,应该考虑如下安全服务:

· 源鉴别: 提供通信对等实体和信息源的论证;

· 访问控制: 禁止通过开放型网络非法获取可利用的资源;

· 数据保密: 对敏感信息进行加密,以防止非法泄漏;

· 不可抵赖: 禁止发送方对发出消息的否认,或接收方对接到消息的蓄意抵赖;

· 数据完整可靠: 对于各种主动攻击,用相应策略恢复。

为了确保这些安全服务机制的实现,计算机密码学和信息的安全保密技术是不可少的工具。

4. EDI 系统安全对策

EDI 应用系统可以看成是信息系统的一部分,因此 EDI 系统的安全策略,就是信息系统安全保密技术的应用。其中包括公开钥加密体制 RSA 和数字签名方法、密钥分配与管理、访问控制权限、认证系统设计、数据信息完整性和设置随机口令等。系统的安全策略,就是信息系统安全保密技术的应用。对于 EDI 系统的安全可以分成几个级别考虑,如图 5.13 所示。

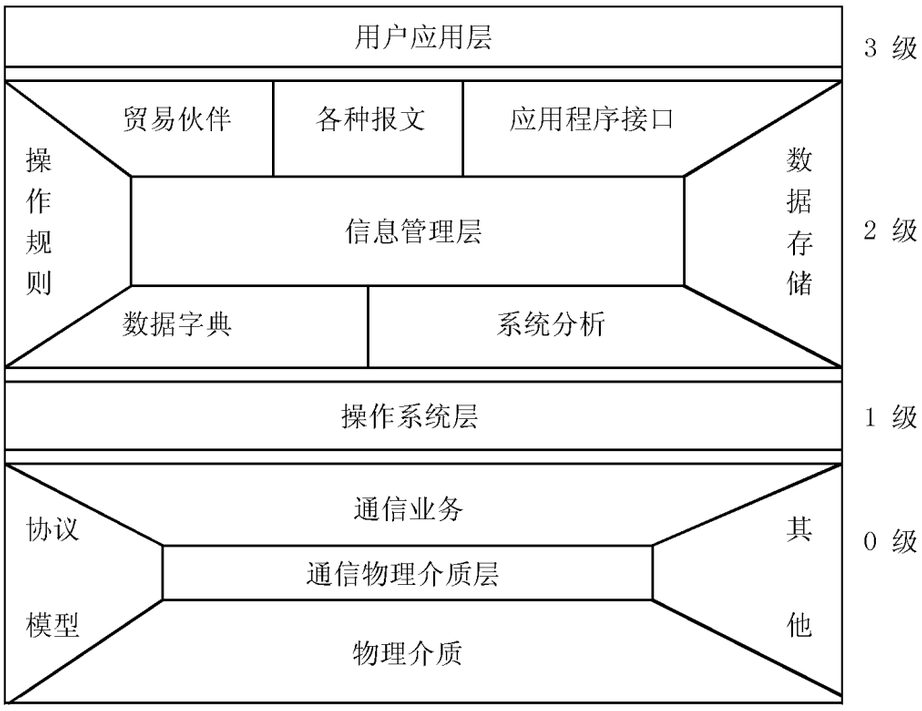


图 5.13 EDI 系统安全级别

1) 通信物理介质层:这层包括具体的通信物理介质,例如: 通信线路、交换设备等。它主要完成具体的通信业务,进行消息的端对端发送,该层定义为 0 级安全级别,是一个接口级的安全,可通过实施网络安全策略,为通信过程的安全奠定基础。

2) 操作系统层: 它介于物理介质层与应用系统层之间,为应用系统方便的访问物理介质而设计的;同时其他用户对本地系统的访问也是通过操作系统层完成的,该层定义为 1 级安全级别。在这层需要身份识别、验证与数字签名,设置随机口令加密手段。

3) 信息管理层:这层包括多个组成单元,核心部分是 EDI 信息处理模块,该层定义为 2 级安全级别,是实现 EDI 系统安全的重要部分。经常用到美国联邦政府的 DES 算法和各种公开钥加密体制,以保证信息管理系统的安全。

4) 用户应用层:该层定义为 3 级安全级别,它只与用户接口相关,它所涉及的安全内容,从信息管理系统之间的合同协议中体现出来。

5. EDI 安全服务实现机制

将信息安全技术应用于 EDI 系统,需要与整个 EDI 系统的实现相配合,根据不同的系统运行环境,可以选择安全策略的一部分在某个应用系统中实现。下面应用信息安全技术,总结一下 EDI 安全服务的实现机制。

1) 访问控制

EDI 安全服务首先要解决好访问控制。访问控制的基本任务是禁止非法用户进入系统,防止合法用户对系统资源的非法使用。访问控制能够对用户进行身份验证,并且决定用户对系统资源的访问权限。安全访问控制服务的实现,可以通过建立访问控制信息库来标识对等实体的访问控制;通过设置口令、设置保密标签、规定试探访问次数等。访问控制机制可以加在通信关联任意端点。

2) 数据完整性

数据完整机制,可以保护数据免遭非法篡改和破坏。数据完整性包括两方面:单个数据单元的完整性;数据单元流的完整性。通常两种服务是同时提供的。实现单个数据单元完整性分为发送和接收两个过程:发送时在数据单元上附上一个分量,这个分量是数据本身的函数,这个分量可能是补充信息,而它本身又可生成密码;接收时用收到的分量与生成的相应分量比较,以确定数据在传输过程中是否被修改。数据单元序列完整性,是指防止数据顺序号错乱和丢失,实现单元序列完整性,可以通过某种排序方法完成。

3) 添加信息流

添加信息流机制,可用来提供各种等级级别的保护,以防止信息被非法统计分析。

4) 验证交换

设置口令实现验证服务,验证信息用发送实体提供口令,并由接收实体检验口令,该机制在验证某实体失败时,则将拒绝建立或中止连接,并向安全保密管理中心报告。数字签名技术在验证服务中起着重要作用,一旦发生纠纷,数字签名有法律保证,可以请公证机关仲裁。

5) 数据审查

安全保密数据审查,是专门检查系统记录和活动情况的独立业务,其目的是测试系统控制是否足够,是否符合现行法规和操作制度,帮助评估损失并推荐修改意见。安全保密数据审查要求将有关安全保密方面的信息记录下来,并且对该数据中获得的信息进行分析生成报告。

6) 信息加密

对于敏感信息，无论是在 EDI 信息处理系统，或者信息通信系统中都应该以密文形式出现。实现信息安全保密服务的机制，是密码学加密解密技术。传统的数据加密标准 DES 算法、公开钥密码体制的 RSA 或有限自动机非对称体制都是很好的信息加密方法。

总之，随着 EDI 在全球的广泛应用，它的安全问题越来越受到人们的关注。EDI 系统的安全不仅需要完备的信息安全保密技术，而且还强烈呼吁国际社会制定出严密的法律程序，以保证 EDI 的顺利发展。

第 6 章　网络的安全与保密

计算机网络的重要功能是资源共享和通信,网络用户对网络中信息的安全和保密问题十分关注。安全性指的是保证数据和程序等资源安全可靠,对资源进行保护,以免受到破坏;保密性主要是指对某些资源或信息,需要加以保密,不允许泄露给别人。

本章首先简单的回顾一下计算机网络协议与模型,然后着重讨论对计算机网络安全的威胁以及各种对策。给出了面向网络系统的加密方法和密钥管理方法,实现网络安全的两种有效技术——防火墙和秘密邮件技术。最后结合面向对象的分布式系统,讲述了实现其安全的认证与加密系统。

6.1　网络安全的威胁与对策

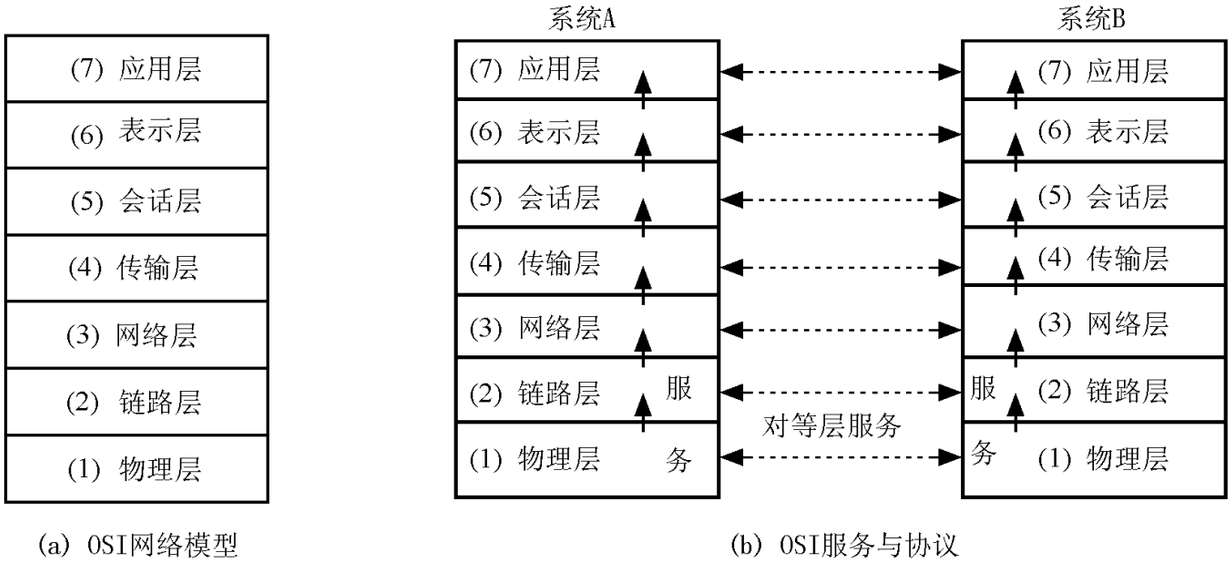
近年来,计算机通信技术飞速发展,信息产业迅速兴起,对社会的经济、文化、生活、国家间的竞争等各个方面产生了巨大影响。特别是国际互联网络 Internet 的使用人数目前急剧增长,估计全世界已经超过 1 亿人。最近连接到 Internet 的企业网用户也很多。企业的重要情报经由 Internet 极易受到攻击。在网络上的电子商业活动日益频繁,电子资金传送的金额数量也在快速增加。但是资源共享和信息安全历来是一对矛盾。据美国联邦调查局统计,美国每年因信息和网络安全问题所造成的损失高达 75 亿美元。1994 年 4 月 16 日,美国金融时报报道,据权威机构统计,平均每 20 秒就发生一起入侵 Internet 计算机的事件。由上述情况看出,信息时代网络安全面对严峻的挑战,安全对策显得尤为重要。网络安全是计算机安全在网络环境下的扩展和延伸,主要包括用户身份验证、访问控制、数据完整性、数据加密、防抵赖和审计追踪等安全要求。防抵赖性安全服务是针对对方抵赖的防范措施,用来证实发生过的操作,可分为对发送方防抵赖,对递交方防抵赖和进行认证。

本节介绍网络模型、协议及安全服务,主要讨论对计算机网络安全的各种威胁,以及相应的对策。加密与签名是最好的技术,用数字信封技术可以快速处理安全保密问题。防火墙与虚拟私人网 VPN(virtual private network) 是保证企业内部网 Intranet 安全的良好措施和工具。

1. 网络模型与协议

由于用户的需要不同,产生了许多种网络系统和网络协议。在同一网络系统中网络的协议是一致的,结点间的通信是方便的,在不同的网络系统中网络的协议可能是不一致的,因此而产生的网络连接和网络之间结点的通信可能会很不方便。为了消除系统之间因协议不同而造成的障碍,使得在互联网范围内,不同的网络系统可以不需要专门的转换

装置就能够通信,1978 年国际标准化组织(ISO)提出了开放系统互联 OSI(open system interconnection)网络设计参考模型。在 OSI 参考模型中,共有 7 个层次,如图 6 .1(a)所示。



(a) OSI网络模型　　(b) OSI服务与协议

图 6 .1　OSI 网络体系

1) 物理层　物理层是设备之间的物理接口,实现比特信号的传输。物理层应保证数据按比特传送的正确性。

2) 数据链路层　数据链路层为网络层提供可靠、无错误的数据信息。

3) 网络层　网络层也是通信子网,是通信子网与网络子层的界面。它主要负责控制通信子网的操作。

4) 传输层　传输层接收来自会话层的数据,将其分成较小的数据单位,送到网络层,实现两层之间数据的透明传送。

5) 会话层　它是用户进入网络的接口,用户必须为这一层与其他机器的进程建立连接而进行协商,按照约定方式处理会话。

6) 表示层　表示层为应用层提供服务,提供数据转换、格式变换、语法选择等功能。表示层协议有文本压缩、安全保密以及虚拟终端等方面的协议。

7) 应用层　应用层是 OSI 的最高层,负责两个应用进程之间的通信,为网络用户之间通信提供专用的程序。

OSI 服务是指在 OSI 模型中,下一层向上一层(自下而上纵向)提供的支持能力,如图 6 .1(b)所示。当 OSI 模型中的第 n 层向 $n+1$ 层提供服务时,需要使用 n 层的功能及 $n-1$层提供的服务。例如,网络层向传输层提供服务时,需要使用网络层本身的功能和数据链路层向网络层提供的服务。

在计算机网络中,任意两结点间的通信规则称为协议。它由一组程序模块组成,又称协议堆栈,每个程序模块在网络通信中有序地完成各自的功能。

OSI 规定所有的网络系统都应具有 OSI 七层模型的功能,网络协议应该可以被映射到 OSI 的每一层。在同一系统中,相同层使用相同的协议;在不同的系统间,对等层使用对等协议。在互联网中,网络协议在对等层提供横向服务。对等层协议的作用是,使位于不同系统的不同协议下的用户能够在对等层上,通过对等层协议进行直接的、透明的、点

到点的对话。

2. 开放互联网络的安全服务

国际标准化组织已于 1989 年对 OSI 开放互联网络环境的安全性进行了深入的研究,在此基础上提出了 OSI 安全体系,定义了安全机制、安全服务、安全管理及有关安全方面的其他问题。此外,还定义了各种安全机制及安全服务在 OSI 中的各层位置。OSI 定义了 11 种威胁,如非法连接、伪装和非授权访问等。对威胁进行安全分析的基础上,规定了以下五种标准的安全服务。

1) 访问控制安全服务

该服务提供一些防御措施,限制用户越权使用资源。访问控制可分为自主访问控制和强制型访问控制两类。实现访问控制安全服务的一种方法就是使用访问控制表,另一种方法就是使用多级访问控制。

2) 对象认证安全服务

该服务用于识别对象的身份或对身份的证实。OSI 环境可提供对等实体身份认证和信源认证等安全服务。对等实体认证是用来验证在某一关联的实体中,对等实体的声明是一致的,它可以确认对等实体身份是否真实。数据信源鉴别是用于验证所收到的数据来源与所声明的来源是否一致,它不提供防止数据中途被修改的功能。

3) 数据保密性安全服务

该服务利用数据加密机制防止信息泄漏,信息加密可以有多种实现方法。

4) 数据完整性安全服务

该服务防止用户使用修改、复制、插入和删除等手段进行非法篡改信息,保证数据具有完整性。

5) 防抵赖性安全服务

该服务是针对对方抵赖的防范措施,用来证实发生过的操作。可分为对发送方防抵赖和对递交方防抵赖,以及发生争执时进行仲裁与公证。

概括安全五要素如下:

(1) 机密性(confidentiality)　保证不向对信息资源没有存取权限的人泄漏信息内容。在网络环境中所采取的对策方法是使用防火墙技术和加密软件。

(2) 一致性(integrity)　从信息资源生成到利用期间保证内容不被篡改。所采取的对策是使用加密软件对信息进行加密。

(3) 可能性(availability)　对于信息资源有存取权限的人,什么时候都可以利用。

(4) 真实性(authenticity)　保证信息资源做成的真实性,具有认证功能。

(5) 责任追究性(accountability)　能够追踪信息资源什么时候被使用,谁在使用及怎样操作使用。

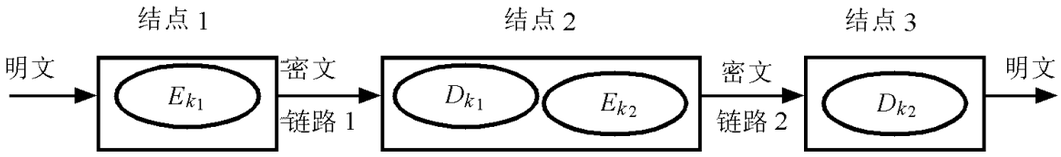
3. 网络通信中的一般加密方式

用户在网络上相互通信,其安全危险来自于非法窃听。有的黑客入侵者通过搭线窃听,截收线路上传输的信息;有的采用电磁窃听,截收无线电传输的信息。因此,对网络传

输的信息进行数据加密,是很有效的安全保密手段,采用一定的加密算法对源信息进行软加密,然后在网络信道上传输密文,这样即使中途被截获,也无法理解信息内容。

1) 链路—链路加密方式

面向链路的加密方法将网络看作由链路连接的结点集合,每一个链路被独立的加密。链路—链路加密方式为两个结点之间通信链路中的信息提供安全性,它与这个信息的起始或终结无关,如图 6.2 所示。每一个这样的链接相当于 OSI 参考模型建立在物理层之上的链路层。



Ek1 及 E_{k_2}为加密变换,D_{k_1}及 D_{k_2}为解密变换。

图 6.2 链路—链路加密示意图

这种类型的加密最容易实现,也很有意义。因为所有的报文都被加密,黑客攻击者无法获得任何关于报文结构的信息,也无法知道通信者、通信内容、通信时间等信息。这可以称之为信号流安全。这种加密方式中,密钥管理相对来说是简单的,只在链路的两站结点需要一个共用密钥。加密是在每个通信链路上独立进行的,每个链路上使用不同的加密密钥。因此,一个链路上的错误不会波及其他链路,影响其他链路上信息安全。

链路—链路信息加密仅限于链路上,不包括结点内部,所以,要求结点本身必须是安全的。另一个较大问题是维护结点安全性的代价。链路—链路加密的优缺点如下。

· 加密对用户是透明的,通过链路发送的任何信息在发送前都先被加密。

· 每个链路只需要一对密钥。

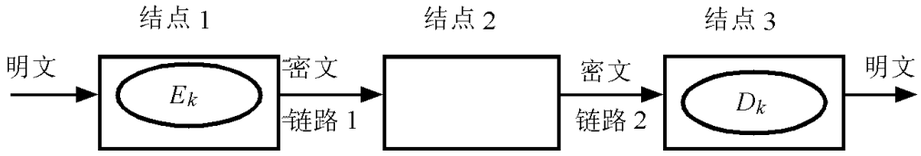
· 提供了信号流安全机制。

· 缺点是数据在中间结点以明文形式出现,维护结点安全性的代价较高。

在链路—链路加密方式中,加密对用户是看不见的、透明的,所有的用户拥有一个设备,加密可以用硬件完成。采用的方式是所有的信息都加密,或任何信息都不加密。

2) 端—端加密方式

端—端加密方法建立在 OSI 参考模型的网络层和传输层。这种方法要求传送的数据从源端到目的端一直保持密文状态,任何通信链路的错误不会影响整体数据的安全性,如图 6.3 所示。对于这种方法,密钥管理比较困难。如果加密在应用层或表示层进行,那么加密可以不依赖于所用通信网的类型。



E_k 为加密变换,D_k 为解密变换。

图 6.3 端—端加密示意图

在端—端加密方式中,只加密数据本身信息,不加密路径控制信息。在发送主机内信息是加密的,在中间结点信息是加密的。用户必须找到加密算法,用户可以选择加密,也可以决定施加某种加密手段。加密可以用软件编程实现。

端—端加密方法将网络看作是一种介质,数据能安全地从源端到达目的端。这种加密在 OSI 模型的高三层进行,在源端进行数据加密,在目的端进行解密,而在中间结点及其线路上将一直以密文形式出现。端—端加密的缺点是允许进行通信量分析,而且密钥管理机制较复杂。

4. 网络安全的威胁及相应的对策

对网络安全的威胁有偶然发生的威胁和故意的威胁两类。偶然发生的威胁如天灾、故障、误操作等。故意的威胁是第三者恶意的行为和电子商务对方的恶意行为。来自恶意的第三者(如外国间谍、犯罪者、产业间谍、职员等)的威胁,如图 6.4 所示。

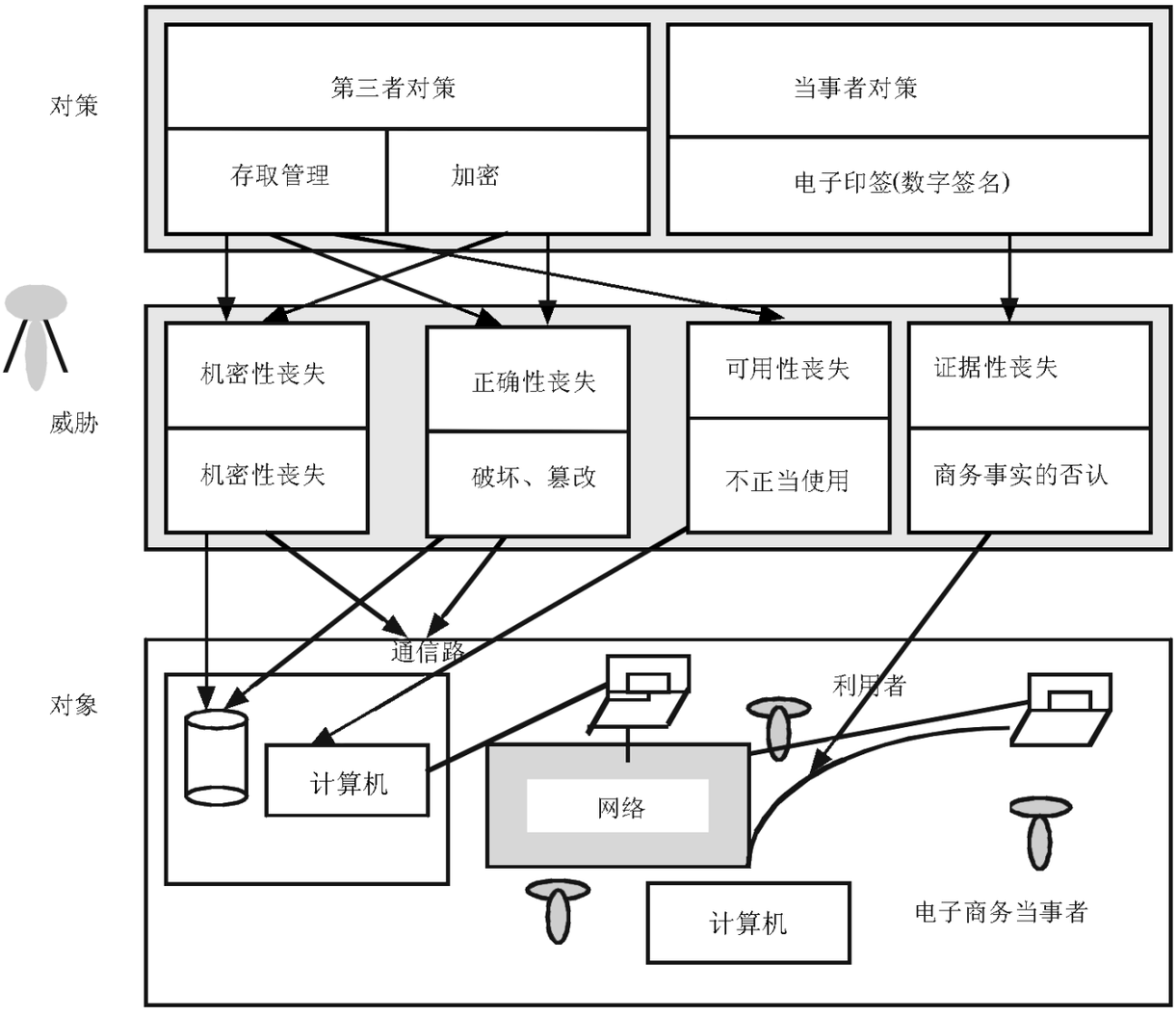


图 6.4 对网络安全的威胁和对策

1) 机密性的丧失: 是指通信线路上和文件内的信息被非法地看见、存取。

2) 正确性的丧失: 是指通信线路上和文件内的信息被不正当地盗听、篡改和破坏。

3) 可用性的丧失: 是指计算机的功能和保存的信息被外部环境的计算机不正确地利用而导致不能使用。

4) 证据性丧失: 是指电子商务事实被否认、伪造和篡改。对于电子商务对方的威

胁,考虑证据性丧失是必要的,如“卖股票 400 股改成卖 4000 股,请赔偿损失”,针对这个不正当的要求,应该能出示证据证明自己的正确性。

5) 受保护系统被非法访问:是指攻击者使用主机探测的结果对目标系统进行攻击,获得对受保护系统的访问权限后,攻击者有多种选择:

· 攻击者可能会安装探测器,包括特洛伊木马程序,用来探测所在系统的活动,收集了一些账户名和口令。攻击者用这些信息将攻击扩展到其他机器。

· 攻击者可能试图毁掉攻击的痕迹,在受损害的系统上建立一个新的安全漏洞或后门,以便在原始攻击被发现后能够继续访问这个系统。

· 攻击者可能会发现对受损系统有信任的主机,他就可以利用某个主机的这种弱点,将整个攻击在整个机构网络上展开。

· 如果攻击者能够在受损系统上获得特别访问权限,他就可以读取邮件、搜索和窃取私人信件,毁掉重要信息数据。

针对上述威胁应该采取相应的安全对策。

安全对策分为直接对策和间接对策。间接对策包括安全监视、安全教育、安全监理等。这里主要讨论直接对策。

1) 对第三者攻击的直接对策

针对第三者的直接的攻击所采取的直接对策就是存取管理和加密两种对策。

(1) 存取管理技术

存取管理防止不正确地存取通信线路上的文件信息,以确保信息安全。如果存取管理控制得好,就不能进行非法攻击,这成为直接的对策。要进行适当的存取管理,用户认证技术和存取控制技术是非常重要的。

用户认证是在自己的计算机向网络存取时实施的技术,用户认证分为三类:利用本人的知识,如密码;利用本人持有物,如磁卡、IC 卡、光卡等;利用本人的身体特征,如指纹、声纹、视网膜模式、DNA 等。

(2) 加密技术

加密技术能达到这样的目的:即使存取控制失败,第三者获得了信息,他也不能理解信息内容。加密是对机密性丧失和正确性丧失两种威胁的有效的对策。

加密方式分为对称钥加密和公开钥加密两种,如表 6 .1 所示。对于对称钥加密方式,加密与解密使用相同的密钥,对称钥加密的典型代表是美国数据加密标准 DES 算法。对于公开钥加密方式,加密与解密使用不同的密钥,公开钥加密方式的典型代表是美国开发的 RSA 算法。DES 算法速度快,适合于大量数据的加密,RSA 适合于秘密钥的传送及数字签名。

表 6 .1　对称钥加密和公开钥加密的比较

比较项目	代表	密钥关系	秘密钥的传送	数字签名	加密速度	主要用途
对称钥加密	DES	加密钥与解密钥相同	不必要	容易	快	数据加密
公开钥加密	RSA	加密钥不同于解密钥	必要	困难	慢	数字签名、密钥分配加密

公开钥加密很方便,但加密与解密速度较慢,这是一个弱点。为解决这一问题出现了

数字信封技术,如图 6 .5 所示,发送者 A 使用数字信封向接收者 B 传送数据。

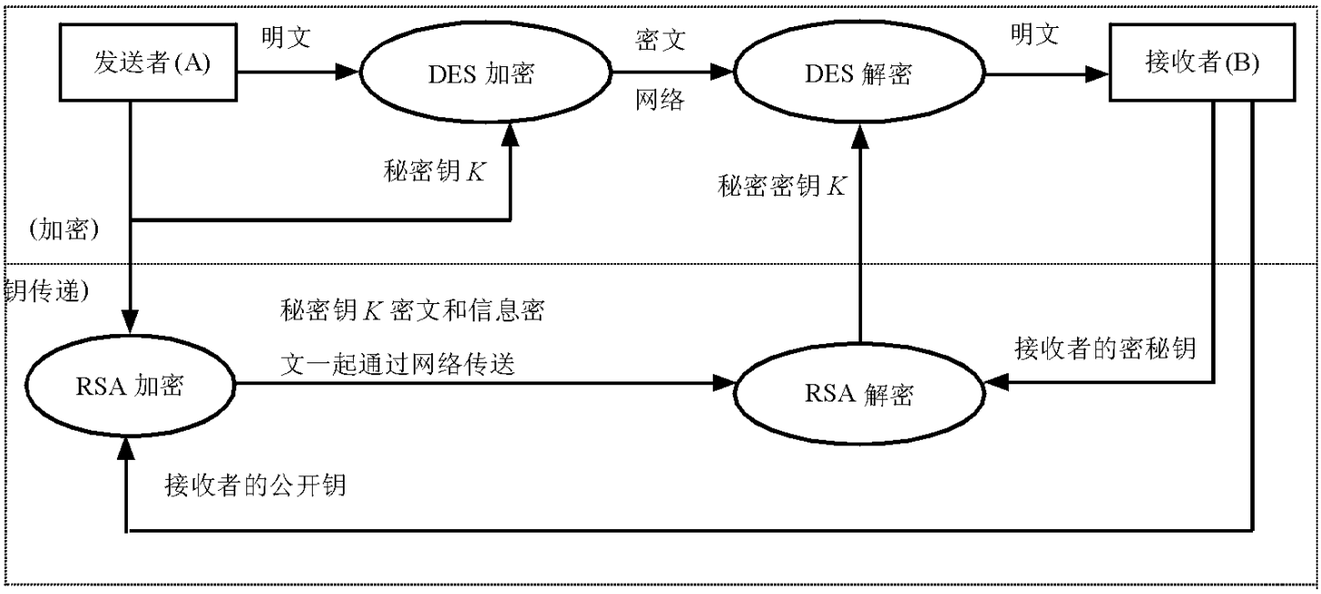


图 6 .5　数字信封技术

首先发信者 A 使用 DES 算法用对称钥 *K* 将明文原信息加密获得密文 ciphertext,然后使用接收者 B 的公开钥将对称钥 *K* 加密获得 Kciphertext,将 Kciphertext 和 ciphertext 一起通过网络传送给 B。B 方接收到密文信息后,首先用 B 的秘密钥解密而获得 *K*,再用 *K* 将 ciphertext 解密而最后获得明文原信息。由此,起到了对明文信息保密的作用。

2) 对电子商务威胁的直接对策

人们一直使用手写签名来证明文件的原作者,或至少证明签名者同意文件的内容。签名是可信服的,但如果签名被伪造,签名之后的文件也能被更改。解决这些问题的对策之一就是利用公开钥加密法进行两种数字签名认证。

功能 1: 用户认证功能,能够证明进行数字签名的用户本人。

功能 2: 信息认证功能,能够证明签了名的信息没有被更改。

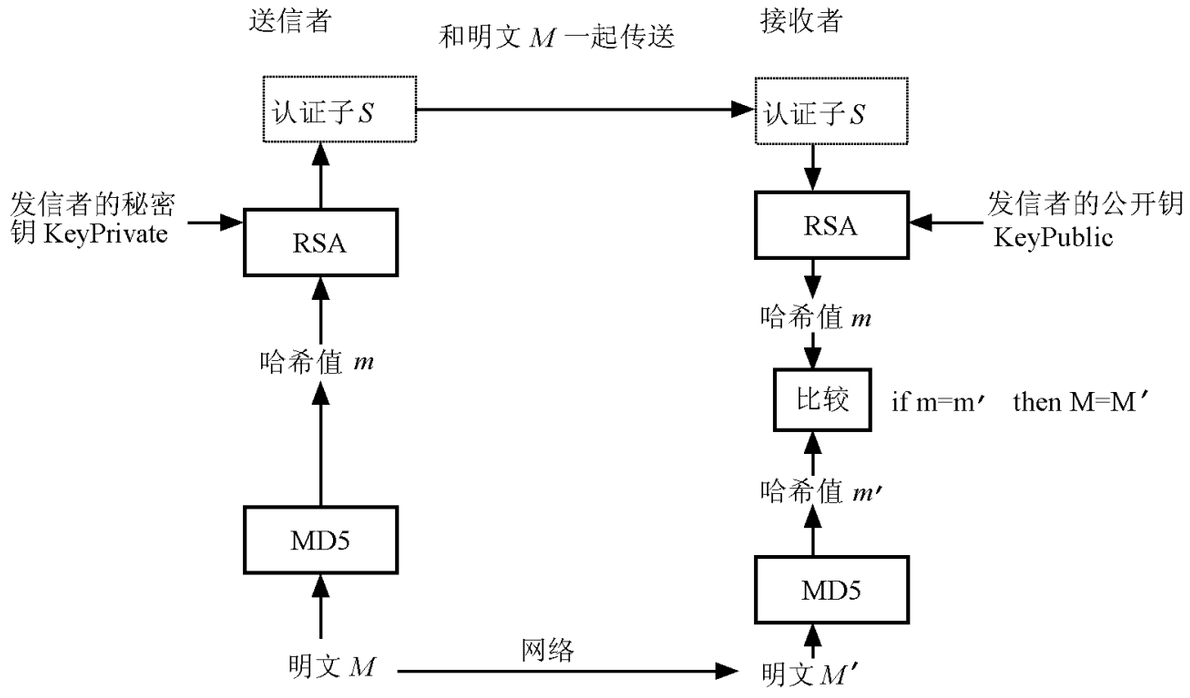


图 6 .6　使用认证子比较法的数字签名

认证的功能如图 6 .6 所示。用认证子比较法进行数字签名。明文 M 用哈希函数 MD5 压缩,进行数据融合获得哈希值 m。将 m 用送信者的秘密钥,使用 RSA 公开钥算法进行加密而获得认证子 S。把 S 和明文 M 一起通过网络传送给接收者。

接收者对接收到的明文 M'用相同的 MD5 计算哈希值 m',同时将认证子 S 用送信者的公开钥使用 RSA 算法解密而获得 m。与 m'比较,如果两者一致,那么说明接收的明文和认证子是正确的,能够判断出没有发生篡改。

6.2 网络系统的密钥管理方法

在加密系统中,算法是很重要的,但一般情况下算法是公开的,秘密全部寓于密钥之中,所以密钥的管理更重要,而且面向网络系统的密钥管理尤为重要。密钥的传统管理方式就是采用密文传送的通信方法以外的安全通信手段将密钥传送。密文是通过网络传送的,密钥可以通过电话或传真传送。本节介绍几种常用的密钥管理方法: Diffie-Hellman 密钥管理方法、基于公开钥加密体系的密钥管理方法和基于 KPS 方法的密钥管理方法。

1. Diffie-Hellman 密钥管理方法

DH 方法是 1976 年 Diffie 和 Hellman 提出的 Diffie-Hellman 密钥交换方法。选择大素数 q 和生成元 α $(1 < \alpha < q)$,其中 q 和 α 是公开的,下面以通信双方 A 和 B 为例,介绍 DH 方法的思想。

A 和 B 分别选择任意的数 X_a 和 X_b 作为各自的秘密信息,首先计算公开的信息 Y_a 和 Y_b:

$$Y_a = \alpha^{X_a} \bmod q$$

$$Y_b = \alpha^{X_b} \bmod q$$

接着 A 和 B 相互地给对方发送 Y_a, Y_b,然后计算共有密钥。

由 A 计算 K_{ab}: $K_{ab} = Y_b^{X_a} \bmod q = \alpha^{X_b X_a} \bmod q$

由 B 计算和 K_{ab}相同的值 K_{ba}: $K_{ba} = Y_a^{X_b} \bmod q = \alpha^{X_a X_b} \bmod q = \alpha^{X_b X_a} \bmod q$

A(或 B)秘密地拥有 X_a(或 X_b),并拥有通信对方的公开信息 Y_b(或 Y_a),就可以计算 K_{ab}(或 K_{ba})。K_{ab}(或 K_{ba})的安全性要得到确保。为了加深理解上面的原理,下面给出 DH 法的一个例子。

假定前面公式中的基本值 α 和 q 赋值为 $\alpha = 5$, $q = 563$,设 A 的秘密信息为 $X_a = 9$,设 B 的秘密信息为 $X_b = 14$。计算 A 和 B 的公开信息分别为 Y_a 和 Y_b:

$$Y_a = \alpha^{X_a} \bmod q = 5^9 \bmod 563 = 78$$

$$Y_b = \alpha^{X_b} \bmod q = 5^{14} \bmod 563 = 534$$

X_a 和 X_b 是秘密的信息, Y_a 和 Y_b 是公开的信息。A 和 B 使用 X_a, X_b, Y_a, Y_b 可以计算共有密钥:

$$K_{ab} = Y_b^{X_a} \bmod q = 534^9 \bmod 563 = 117$$

$$K_{ba} = Y_a^{X_b} \bmod q = 78^{14} \bmod 563 = 117$$

在这里,A 和 B 的共有密钥是 117,就可以进行加密通信了。

采用 Diffie-Hellman 密钥管理方法,根据各自的秘密信息 X,计算公开信息 Y,之后,把各自的 Y 传送给对方,再计算共有密钥。但这个过程是在密文数据信息传送前必须做的,这意味着在密文信息传送前事先要进行通信。使用 Diffie-Hellman 密钥管理方法的通信过程如表 6 .2 和图 6 .7 所示。

表 6 .2　使用 Diffie-Hellman 密钥管理方法实现加密的过程

	发信者 A	受信者 B
前期准备	秘密信息 X_a	秘密信息 X_b
	公开信息 $Y_a = \alpha^{X_a} \bmod q$	公开信息 $Y_b = \alpha^{X_b} \bmod q$
事前通信	取得 B 的公开信息 Y_b	取得 A 的公开信息 Y_a
共有密钥生成	计算 $K_{ab} = Y_b^{X_a} \bmod q$	计算 $K_{ba} = Y_a^{X_b} \bmod q$
会话钥生成	由随机数生成器生成会话钥	
数据信息加密	用会话钥将明文信息加密	
会话钥被加密	用 K_{ab} 将会话钥加密	
密文送信	数据密文和加密后的会话钥一起送给 B	接收从 A 发送来的数据密文和加密后的会话钥
解密会话钥		用 K_{ba} 将会话钥密文解密获得会话钥明文
数据信息解密		用会话钥将数据信息密文解密获得数据明文

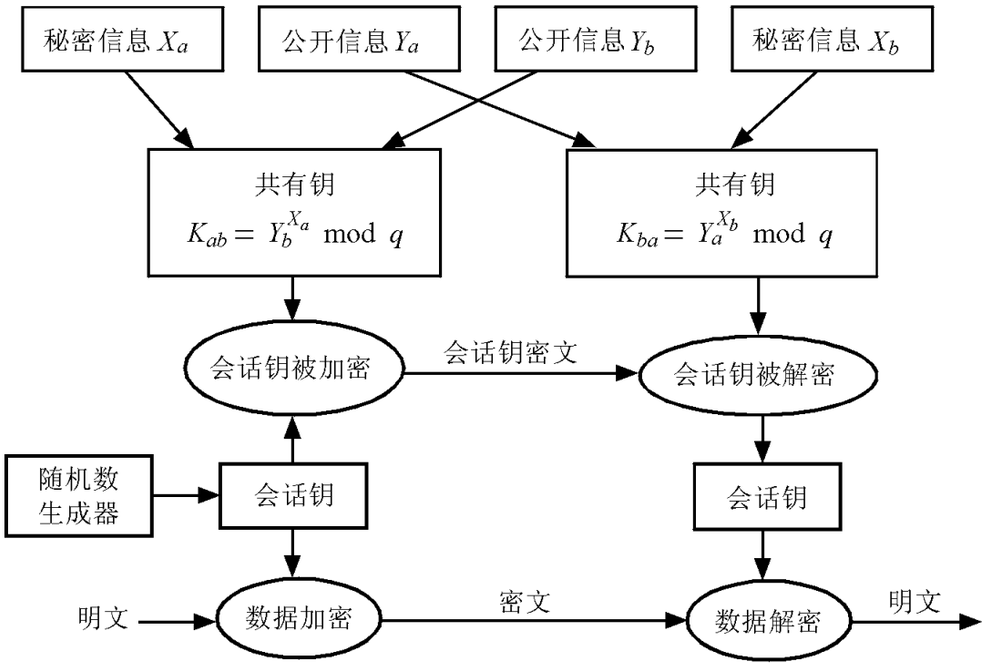


图 6 .7　Diffie-Hellman 密钥管理方法

2. 基于公开钥加密体制的密钥管理方法

1) 需要的密钥对

需要的密钥对是 DH 方法的两倍。受信者 B 的秘密钥和公开钥分别为 K_h, K_q,其中秘密钥 K_h 只有受信者 B 拥有,公开钥 K_q 是公开的。对于公开钥加密方式, $K_h \neq K_q$。A 从 B 一方得到公开钥,处理过程如表 6.3 和图 6.8 所示。

表 6.3　基于公开钥加密体制的密钥管理方法

	发信者 A	受信者 B
密钥对生成		生成秘密钥 K_h,公开钥 K_q
事前通信	取得 B 的公开钥 K_q	
会话钥生成	由随机数生成器生成会话钥 K_c	
数据信息加密	用 K_c 将明文信息加密	
会话钥被加密	利用 K_q 把 K_c 加密形成 K_s $K_s = E(K_q, K_c)$	
密文送信	数据密文和加密后的会话钥 K_s 一起送给 B	接收从 A 送来的数据密文和加密后的会话钥 K_s
解密会话钥		用 K_h 将 K_s 解密获得会话钥明文 K_c
数据信息解密		用会话钥 K_c 将数据信息密文解密获得数据明文

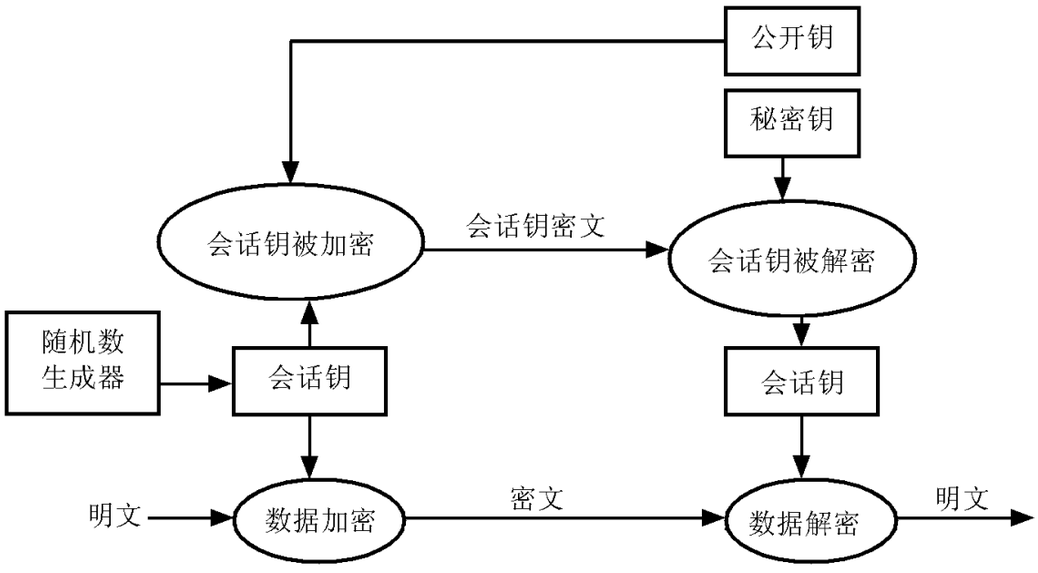


图 6.8　基于公开钥加密体制的密钥交换

使用 K_q 对会话钥明文 K_c 进行加密(*E* 变换)形成密文 K_s。使用 K_h 对 K_s 解密(*D* 变换)形成会话钥明文 K_c,过程表示如下:

$$K_s = E(K_q, K_c)$$

$$K_c = D(K_h, K_s)$$

用公开钥对会话钥进行加密,用会话钥对数据加密,两部分密文内容一起送给 B 方。B 方使用自己的秘密钥将会话钥密文解密,以获得会话钥明文,用会话钥将数据密文解密,获得数据明文信息。

2) 伪造篡改问题

使用公开钥加密方式的密钥管理问题之一,就是伪造篡改问题。使用公开钥进行密钥管理的场合,从对方直接取得受信者的公开钥,密钥服务器是必需的。这时,产生怎样的疑问呢?取得的公开钥是自己通信对方的公开钥吗?第三者(盗听者)会不会篡改呢?并且,接收密文时,密文是谁生成发出的?送信者是谁?受信者是谁?这就是伪造篡改问题。

解决这一问题的手段就是向系统中加入认证证书 CA(certification authority)机制。送信者取得受信者公开钥时,公开钥具有认证证书作的签名。并且,送信者送信时也要签名。这种方法的前提是假定认证证书是可信赖的。使用从认证证书发行的签了名的公开钥,就可以防止伪造篡改问题,而认证证书的伪造篡改问题怎样防止呢?这可以采用层次性认证证书机制来解决。关于认证证书 CA 机制在本章的秘密电子邮件一节会详细介绍。

3. 基于 KPS(key predistribution system,密钥预分配系统)的密钥管理方法

1) 基本原理

给出系统参加者的公开的认证子 ID,向系统中心准备随机构成的秘密数据 G。把发信者 A 的认证子定义为 ID_a 时,计算 $X_a = G^{ID_a}$。把这个计算结果 X_a 称为 A 的秘密算法。把接收者 B 的认证子定义为 ID_b,计算 $X_b = G^{ID_b}$。把这个计算结果 X_b 称为 B 的秘密算法。A,B 双方通信时,把各自的认证子传递给对方,由对方的认证子及自己的秘密算法生成密钥,A 方的共有密钥为:

$$K_{ab} = X_a^{ID_b} = (ID_a \cdot G) \cdot ID_b = ID_a \cdot G \cdot ID_b$$

B 方的共有密钥为

$$K_{ba} = X_b^{ID_a} = (ID_b \cdot G) \cdot ID_a = ID_b \cdot G \cdot ID_a$$

其中,$ID_a \cdot G \cdot ID_b = ID_b \cdot G \cdot ID_a$,也就是 A 和 B 有相同的共有密钥。这种方法叫作线性模式。

2) 线性模式

1986 年,松本努教授和今井秀树教授提出了一种线性模式。其主要原理介绍如下:

首先把表示各用户的文字信息变成 n 比特的数据。对应单向函数 h 和密钥长度 k 比特,定义单向函数 g。在管理 KPS 的中心,准备随机的 k 个 $n \times n$ 行列式:$M_1, M_2, \cdots, M_k$ 各行列元素值必须是 0 或 1,而且,各个行列式必须关于对角线对称。

这样的对称矩阵 $M_1, M_2, \cdots, M_k$ 相当于基本原理一节中的 G。

例:假设 $n = 4$, $k = 3$ 管理中心的对称矩阵 M_1, M_2, M_3 设定如下:

$$M_1 = \begin{matrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{matrix}$$

$$M_2 = \begin{matrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{matrix}$$

$$M_3 = \begin{matrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{matrix}$$

从 M_1 到 M_3 构成了 KPS 方式的基本核心。

A 和 B 进行加密通信。首先两人将电子邮件地址的认证子向中心登录,将认证子用单向函数变成 n 个比特。A 和 B 的认证子用单向函数变成 n 个比特后的结果表示为 ID_a, ID_b:

$$ID_a = h(A) = (0101)$$

$$ID_b = h(B) = (0011)$$

在登录中心里,首先从 M_1 到 M_3 与 ID_a 作乘法运算,从而计算出 V_{a_1} 到 V_{a_3}。这种计算方法与一般行列式中计算方法相同,相关的加运算使用异或原理。这里以 A 为例,因为 A 的 ID_a 的第 1 列和第 3 列值为 1,与 M_1 的第 1 列和第 3 列进行异或求和,得

$$V_{a_1} = M_1\ ID_a = (1001)$$

$$V_{a_2} = M_2\ ID_a = (1000)$$

$$V_{a_3} = M_3\ ID_a = (1110)$$

3) 抗干扰模型

得到的结果 V_{a_1} 到 V_{a_3} 在基本原理上同秘密算法的 X_a。将单向函数 h 和 g 及秘密算法(V_{a_1} ～ V_{a_3})装入抗干扰模型,传给 A 方。抗干扰模型具有这样的特点: 对模型内的信息很难进行不正当的读取或篡改。这个模型内的个人信息(V_{a_1} ～ V_{a_3}),A 自己也看不见。对于 B 方,用同样的方法计算秘密算法 X_b(V_{b_1} V_{b_3})时,得到

$$V_{b_1} = M_1\ ID_b = (0011)$$

$$V_{b_2} = M_2\ ID_b = (1110)$$

$$V_{b_3} = M_3\ ID_b = (1001)$$

把这些装入抗干扰模型,传递给 B。

从 A 向 B 通信的场合,A 方要在抗干扰模型中进行下面的工作。

(1) 把 B 的认证子输入给单向函数 h,单向函数 h 对全部用户有相同的定义。所以得出 ID_b = (0011)

(2) 得到的 B 的结果(0011)和 A 的秘密信息的一个 V_{a_1} 相乘。此时加法运算规则使用异或原理 $ID_b * V_{a_1} = 0*1 + 0*0 + 1*0 + 1*1 = 1$

(3) 同理把 ID_b 和 V_{a_2}, V_{a_3} 作相乘计算。

(4) 把得到的结果按顺序进行排列,形成 k_{ab} = (1 0 1)

(5) 借助于单向函数 g 计算得到的 k_{ab} 是共有密钥。

在 B 一方,把 A 的认证子通过单向函数的计算得到 ID_a = (1 0 10)。然后同样地进行 V_{b_1} 到 V_{b_3} 与 ID_a 的计算。

$$ID_a * V_{b_1} = 1\times0 + 0\times0 + 1\times1 + 0\times1 = 0 + 0 + 1 + 0 = 1$$

$$ID_a * V_{b_2} = 1\times1 + 0\times1 + 1\times1 + 0\times1 = 1 + 0 + 1 + 0 = 0$$

$$ID_a * V_{b_3} = 1\times1 + 0\times0 + 1\times0 + 0\times1 = 1 + 0 + 0 + 0 = 1$$

所以 $k_{ba} = (1\ 0\ 1)$，k_{ab}和 k_{ba}是相等的，是共有密钥。

实际应用时，使用会话密钥将明文信息加密。会话密钥是通过随机数生成器生成的，对于会话密钥是用共有密钥加密的。将会话密钥密文与数据信息密文一同从 A 方发送给 B 方。B 方用共有密钥将会话密钥密文解密得到会话密钥明文，再用会话钥明文对数据密文信息解密获得所需的数据明文。基于 KPS 的密钥管理如表 6 4 和图 6 9 所示。

表 6 4　基于 KPS 方式的密钥管理过程

	A	B
事前分配	取得对应 A 认证子的秘密算法	取得对应 B 认证子的秘密算法
会话钥生成	由随机数生成器生成会话钥	
数据信息加密	用会话钥将明文信息加密	
共有密钥生成	把 B 方的认证子输入给秘密算法，生成共有密钥	把 A 方的认证子输入给秘密算法，生成共有密钥
会话钥被加密	利用共有密钥把会话钥加密形成密文	
密文送信	数据密文和加密后的会话钥一起送给 B	接收从 A 送来的数据密文和加密后的会话钥
解密会话钥		用共有密钥对会话钥密文解密，获得会话钥明文
数据信息解密		用会话钥将数据信息密文解密获得数据明文

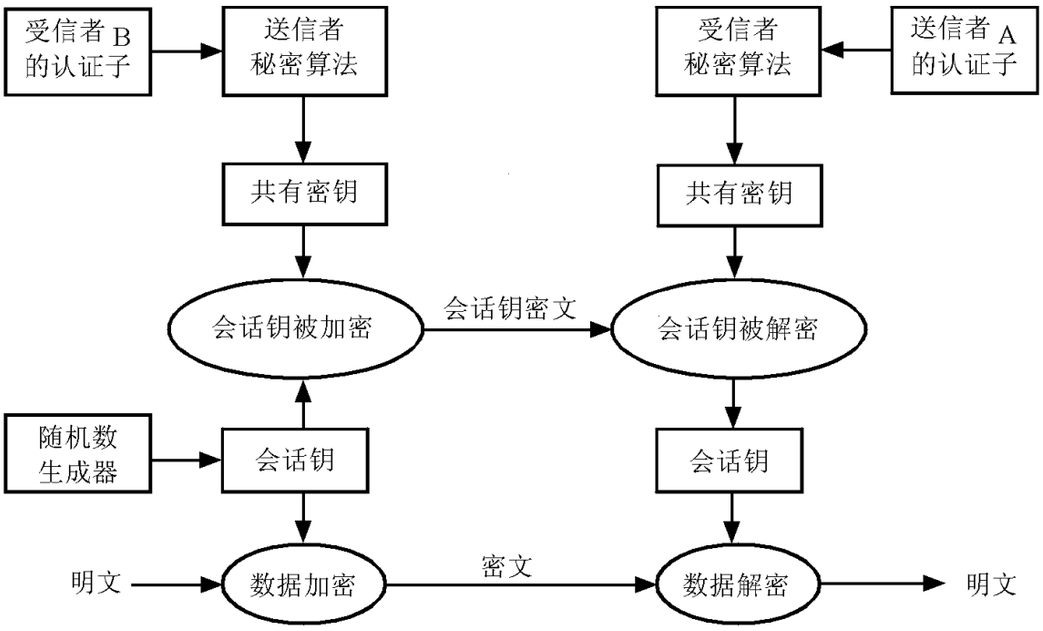


图 6 9　基于 KPS 方式的密钥管理

像这样使用 KPS 方式的场合,相互通信双方根据自己所特有的秘密算法进行认证,像公开钥加密方式中的伪造欺诈现象基本上得到杜绝。

6.3 Internet 安全与防火墙技术

Internet(因特网)是由美国开发研制的世界上规模最大的计算机互联网络。互联网又称为网络的网络。Internet 提供了多种服务,它是开放的不安全的互联网,Internet 上的信息极易受到攻击和破坏,它的安全与保密很重要。本节讨论 Internet 的安全对策及防火墙技术,介绍了紧急对应组织 IRT(incident response team)。防火墙技术用于在开放的不安全的 Internet 环境下构筑一个相对安全的网络环境,满足安全要求。

1. Internet 服务及安全对策

1) Internet 主要服务

Internet 的主要服务包括基本服务和应用服务,具体内容描述如下。

(1) 电子邮件 E-mail 服务

电子邮件 E-mail 服务是指计算机之间通过网络传输信件、文本、图形、图象等多媒体信息。电子邮件提供了一种简便、快捷的现代通信方式,加速了信息的交流与传递,缩短了人与人之间的距离。

(2) 文件传输服务 FTP(file transfer protocol)

文件传输服务是在 Internet 上实现用户将文件从一台计算机传送到另一台计算机上的服务。

(3) 远程登录服务 Telnet

远程登录是指通过 Internet 网把本地计算机连接到一台远程分时系统计算机上,使本地计算机完全成为对方主机的远程仿真终端用户。

(4) 网络新闻服务 USENET NEWS

网络新闻系统包括新闻稿、新闻组、新闻服务器、新闻阅读服务器,用户在使用时只要与一台新闻服务器连通,就可以阅读网络新闻。

(5) WWW (World Wide Web)服务

电子邮件、文件传输、远程终端注册、网络新闻服务在 Internet 早期已经很是流行,而 WWW 是一种全新的基于 Internet 的服务,WWW 是一个专用术语,用于描述 Internet 上的所有可用信息和多媒体资源。用户可以使用被称作 Web 浏览器的应用程序访问这些信息。Web 浏览器可用于搜索、查看和下载 Internet 上的各种信息。

上述讨论的是 Internet 的基本服务,Internet 的应用服务包括电子购物、电子决策和信息提供服务。对应这些服务要采取一定的安全对策。

2) Internet 的安全对策

(1) 电子邮件 E-mail 的安全对策

实现电子邮件 E-mail 安全的手段就是加密和签名。作为 Internet 标准而提出的 PEM(privacy enhanced mail,秘密电子邮件)和美国 Zimmerman 开发的 PGP 是实现 E-

mail 安全的两个有代表性的策略。PEM 协议提供加密、验证、信息完整性保护和密钥管理功能。PEM 将在 6.6 节中详细介绍。

(2) WWW 服务器和客户间的安全对策

SHTTP(secure HTTP)和 SSL(secure socket layer)是两个有名的 WWW 服务器和客户间的安全对策。SHTTP 是 EIT(Enterprise Integration Technologies)公司开发的面向 WWW 的安全协议,HTTP(hyper text transfer protocol) 是加入了加密功能的协议。但是 SHTTP 是 WWW 服务器和 WWW 客户间专用的。

SSL 是美国的 Netscape 公司开发的面向 TCP(transmission control protocol)应用的通信路加密协议,已安装在 TCP 协议和 HTTP,FTP 等的应用协议间。SSL 不仅适合于 WWW 服务器和客户间通信路的加密,而且也适合于 FTP 和 Telnet 等的数据获取的通信路的加密,并且服务器具有认证的功能。最近使用 SSL 的用户比较多。

(3) 为实现应用服务的加密和签名技术

为了实现电子购物、电子决策的电子商业活动的安全,加密技术和数字签名技术是非常必要的。这个功能已经以 SET(secure electronic transactions,安全的电子商务)的形式标准化。SET 已运用于实际系统进行实验,并且为了提供信息情报服务,著作权管理的功能是必要的。对于著作权管理要防止不正当的拷贝,其中加密和数字签名技术被广泛的使用。

(4) 把 Internet 接到企业信息网 Intranet 时要保证企业信息网的安全性

可采用防火墙技术保护企业信息网。在流出入通路中的一个场所集中地监视出入的信息,防止不正当的侵入,只允许被授权的信息通过。在 Internet 和 Intranet 之间设置防火墙,会提高网络的安全性,使用虚拟私人网会增加网络的保密性。

3) 紧急对应组织 IRT

美国 1988 年以“Internet Worm”事件为契机,作为 Internet 安全相关的情报中心,设立了 CERT/ CC(Computer Emergency Response Team Coordination Center),收集 Internet 上发生的安全问题的情报,分析其原因,促进对策的开发,并且提供对被害者的咨询。像 CERT/ CC 这样的面向 Internet 安全的紧急对应组织称作 IRT(Incident Response Team)。

IRT 针对 Internet 通信安全应提供以下功能。

(1) 收集情报　收集发生的安全问题情报,进行分类,确定被害范围。根据收集的情报,确定目前安全问题的特征。

(2) 分析原因　用未知的手段进行不正当的侵入发生的场合,为避免用相同手段再发生类似的安全问题,必须分析其原因。进一步地有必要评价这种手段所具有的影响。但是如果是 Internet 团体使用的软件的问题的话,对 Internet 团体影响是大的。具有广泛影响的手段存在的场合,对 Internet 团体有必要迅速地发出警告。

(3) 和企业的协作　现在利用的软件大部分是商用软件包。为此对不正当侵入的原因确定和对策的研究,软件开发企业必须协力支持。

(4) 情报的流通　针对现在流行的安全问题的情报和警告,所研究的对策情报广泛地流通。面向 Internet 安全的一致努力是 IRT 进行活动的目的,并且管理者和一般利用者要有较高的安全意识,这对保障安全是非常重要的。

(5) 对被害者的咨询　对受到不正当侵入等的被害者,提供对策情报,对管理作业提供一些建议,防止将来安全问题的再发生。为此对被害者提供一些咨询,IRT 应起很大作用。

进入 20 世纪 90 年代以来安全问题频繁发生,各国相继设立了 IRT。如澳大利亚的 AUSCERT、日本的 JPCERT/ CC,北美、欧洲、亚洲各国也建立了自己的国家 IRT(National IRT)。多国籍企业和政府组织规模大,为了迅速对付其组织内的安全问题,开始建立了组织内 IRT,称作 Organizational IRT。我国也建立了自己的紧急对应组织 IRT。

2. 防火墙的概念与体系结构

为了确保信息安全,避免对网络的威胁与攻击,防止对网络资源的不正当的存取,保护信息资源而采取的一种手段就是设置防火墙。

所谓防火墙有理论上的概念和物理上的系统两个含义。理论上防火墙概念指的是提供对网络的存取控制功能,保护信息资源,避免不正当的存取,如图 6 .10 所示。从物理上的系统解释,防火墙是 Intranet 和 Internet 间设置的一种过滤器、限制器。

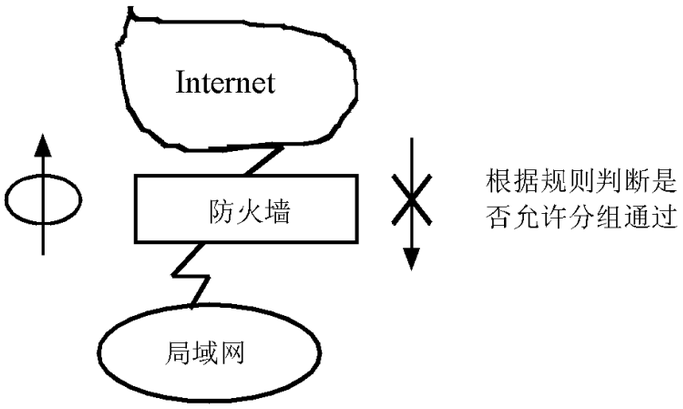


图 6 .10　防火墙的理论概念

目前,有多种防火墙制品销售,具体说哪个制品最好,这很难下结论,要把握网络的利用形态及需要怎样的安全功能,来设计怎样的最优的防火墙。防火墙的分类如表 6 .5 所示。

表 6 5　防火墙的分类

根据连接方式分类	分组过滤型
	应用网关型(或代理服务器)
	电路层网关型
根据构成方式分类	有屏蔽的子网型
	多宿主机型
	堡垒主机型

对应于表 6 .5 中的防火墙的构成分类,相应的防火墙的体系结构分别如图 6 .11,6 .12,6 .13 所示。

有屏蔽子网型防火墙是防火墙的基本类型(图 6 .11) ,它的构成包括 ISP(internet

service Provider,Internet 服务供应商)路由和分组过滤路由,这种防火墙接续在 Internet 和企业局域网之间的干涉地带 DMZ(demi militarized zone)。

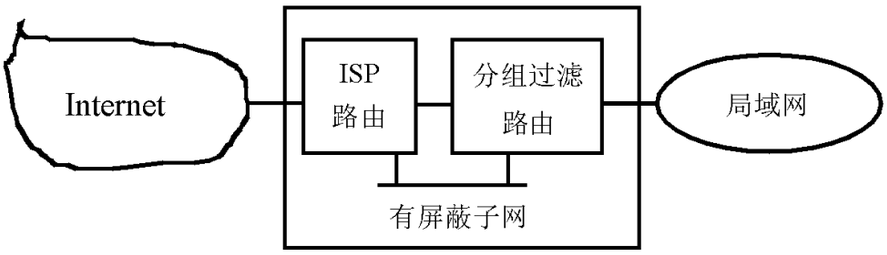


图 6.11　有屏蔽子网型防火墙

在 Internet 和企业局域网之间接续多宿主机,作为代理中继,可以构成多宿主机型防火墙,如图 6.12 所示。多宿主机指的是至少具有两个网络接口的通用计算机。可以利用多宿主机建立防火墙,将多宿主机的一部分端口与 Internet 连接,另一部分端口与企业网连接,同时屏蔽 TCP/ IP 的信息传递功能。在 Internet 和企业网之间禁止信息直接流通,流通的信息要经过防火墙的控制和检查。

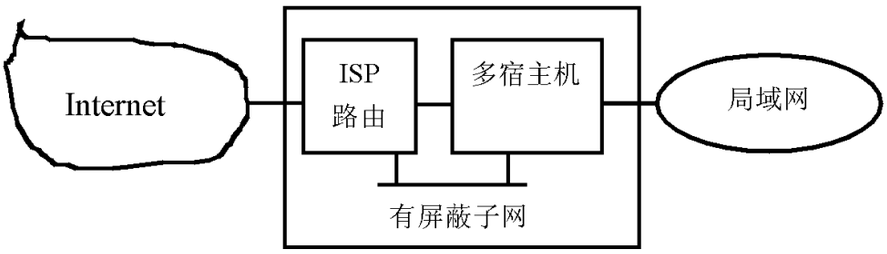


图 6.12　多宿主机型防火墙

有屏蔽子网限制来自外部的存取,对堡垒主机的攻击不影响内部网。堡垒主机与分组过滤路由组合成功能较强的防火墙,如图 6.13 所示。堡垒主机指的是企业网中暴露给 Internet 从而受到来自 Internet 攻击的机器,它容易受到攻击,所以需要加强对它的安全保护措施来防止对它的攻击。堡垒主机通常是防火墙的一个构件,应该具有高度的安全性。

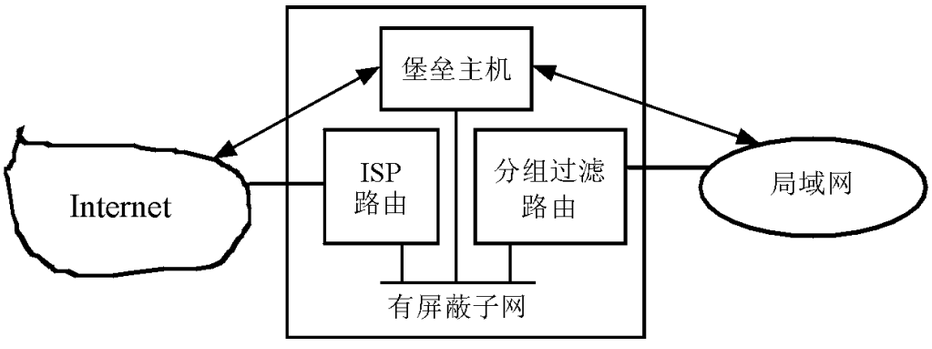


图 6.13　堡垒主机型防火墙

3. 防火墙的优点与用途

目前的防火墙有很多优点,但也有一定的缺点。理想的防火墙应该具有高度安全性、高度透明性及良好的网络性能,而这些特性本身相互制约、相互影响。因此用户可根据实际情况需要,确定、选择使用哪种途径来设计满足自己网络安全需要的防火墙。防火墙发

展方向之一就是设计融合分组过滤和代理(proxy)范围优先的新型防火墙。

(1) 防火墙可以作为内部网络安全屏障。

(2) 防火墙限制了企业网 Intranet 对 Internet 的暴露程度,避免 Internet 网的安全问题对 Intranet 的传播。

(3) 防火墙是设置网络地址翻译器 NAT(network address translator)的最佳位置。Internet的发展突飞猛进,目前使用的网际协议 IP(internet protocol)地址资源发生了地址枯竭危机,NAT 是应付这种危机的有效方法之一。

Internet 防火墙是这样的系统(或一组系统):它能增强机构内部的安全性。防火墙系统决定了哪些内部服务可以被外界访问;外界的哪些人可以访问内部的哪些可以访问的服务,以及哪些外部服务可以被内部人员访问。要使一个防火墙有效,所有来自和去往 Internet 的信息都必须经过防火墙的过滤、检查及存取控制,并且防火墙本身也必须能够免于渗透。但遗憾的是,防火墙系统一旦被攻击者突破或迂回,就不能提供任何的保护了。

Internet 防火墙负责管理 Internet 和内部网络之间的访问。在没有防火墙时,内部网络上的每个结点都暴露给 Internet 的其他主机,极易受到攻击。这就意味着内部网络的安全性要由每一个主机的坚固程度来决定。

Internet 允许网络管理员定义一个中心“扼制点”来防止黑客入侵者、网络破坏者等进入内部网络。在防火墙上很容易监视网络的安全性,并产生报警。网络管理员必须审计和记录所有通过防火墙的重要信息。

4. 防火墙的设计

1) 基本的防火墙设计

在设计 Internet 防火墙时,网络管理员必须作出几个决定:

· 防火墙的姿态

· 机构的整体安全策略

· 防火墙的经济费用

· 防火墙系统的组件和构件

下面介绍前两部分内容。

(1) 防火墙的姿态　防火墙的姿态从根本上阐述了一个机构对安全的看法。Internet 防火墙可能会扮演两种截然相反的姿态:

· 允许没有特别拒绝的任何事情。这种姿态假定防火墙应该转发所有的信息,任何可能存在危害的服务都应在 case-by-case 的基础上关掉。这种方案建立的是一个非常灵活的环境,能提供给用户更多的服务。缺点是,由于将易使用这个特点放在了安全性的前面,网络管理员处于不断的响应当中,因此,随着网络规模的增大,很难保证网络的安全。

· 拒绝没有特别允许的任何事情。这种姿态假定防火墙应该阻塞所有的信息,而每一种期望的服务或应用都是实现在 case-by-case 的基础上。这是一个受推荐的方案。其建立的是一个非常安全的环境,因为只有审慎选择的服务才被支持。当然这种方案也有缺点,就是不易使用,因为限制了提供给用户的选择范围。

(2) 机构的安全策略　如前所述,Internet 防火墙并不是独立的,它是机构总体安全策略的一部分。机构总体安全策略定义了安全防御的方方面面。为确保成功,机构知道其所保护的是什么。安全策略必须建立在精心进行的安全分析、风险的评估以及商业需求分析基础之上。如果机构没有详细的安全策略,无论如何精心构建的防火墙都会被绕过去,从而整个内部网络都暴露在攻击面下。

机构能够负担起什么样的防火墙?简单的分组过滤防火墙的费用最低,因为机构至少需要一个路由器才能连入 Internet,并且分组过滤功能包括在标准的路由器配置之中。商业的防火墙系统提供了附加的安全功能,具体的价格要看系统的复杂性和要保护系统的数量。如果一个机构有自己的专业人员,也可以构建自己的防火墙系统,但是有开发时间和部署防火墙系统等费用问题。还有防火墙系统需要管理、维护等,这些都要增加费用。

应给予特别注意的是,Internet 防火墙不仅仅是路由器、堡垒主机或任何提供网络安全的设备的组合,它也是安全策略的一个部分。安全策略建立了全方位的防御体系来保护机构的信息资源。所有可能受到网络攻击的地方都必须以同样安全级别加以保护。仅设立防火墙系统,而没有全面的安全策略,那么防火墙就形同虚设。

(3) 防火墙系统的组件　在确定了防火墙的姿态、安全策略以及经费预算问题之后,就能够确定防火墙系统的特定组件。典型的防火墙由一个或多个构件组成:分组过滤路由器、应用网层关(或代理服务器)和电路层网关。

2) 基于分组过滤的防火墙设计

(1) 过滤的项目

IP 分组过滤是通过路由的分组过滤。把通过的分组信息头部的内容与设定于路由的存取控制规则进行比较,进行交通存取控制。通常,作为过滤的项目有:IP 地址(源地址、目的地址)、协议(TCP,UDP,ICMP)和接口(源接口、目的接口)。其中,UDP 为 user datagram protocol(用户数据报协议)的缩写,ICMP 为 internet control message protocol(网际控制报文协议)的缩写。

(2) 规则的设定例子

过滤规则的设定方法因路由不同而不同。图 6.14 简单说明了面向网络环境的路由的过滤规则,表 6.6 表示了向 Internet 一侧的接口 s_0 输入的分组过滤规则,表 6.7 设定了局域网 LAN(local area network)一侧的接口 e_0 的分组过滤规则。进一步的,在内部网和 Internet之间直接接续 IP 的场合,内部网的信息(IP 地址)向 Internet 流入,受攻击的可能性不大。

表 6.6　向 s_0 输入过滤规则的例子

序号	动作	协议	源地址	源接口	目的地址	目的接口	说　明
1	许可	UDP	任意	53	172.16.128.10	53	向 DNS 服务器的查询
2	许可	TCP	任意	>1023	172.16.128.10	53	向 DNS 服务器的查询
3	许可	TCP	任意	>1023	172.16.128.10	25	向 SMTP 服务器的存取
4	许可	TCP	任意	>1023	172.16.128.20	80	向 WWW 服务器的存取

表 6.7　以太网接口 e_0 输入过滤规则的例子

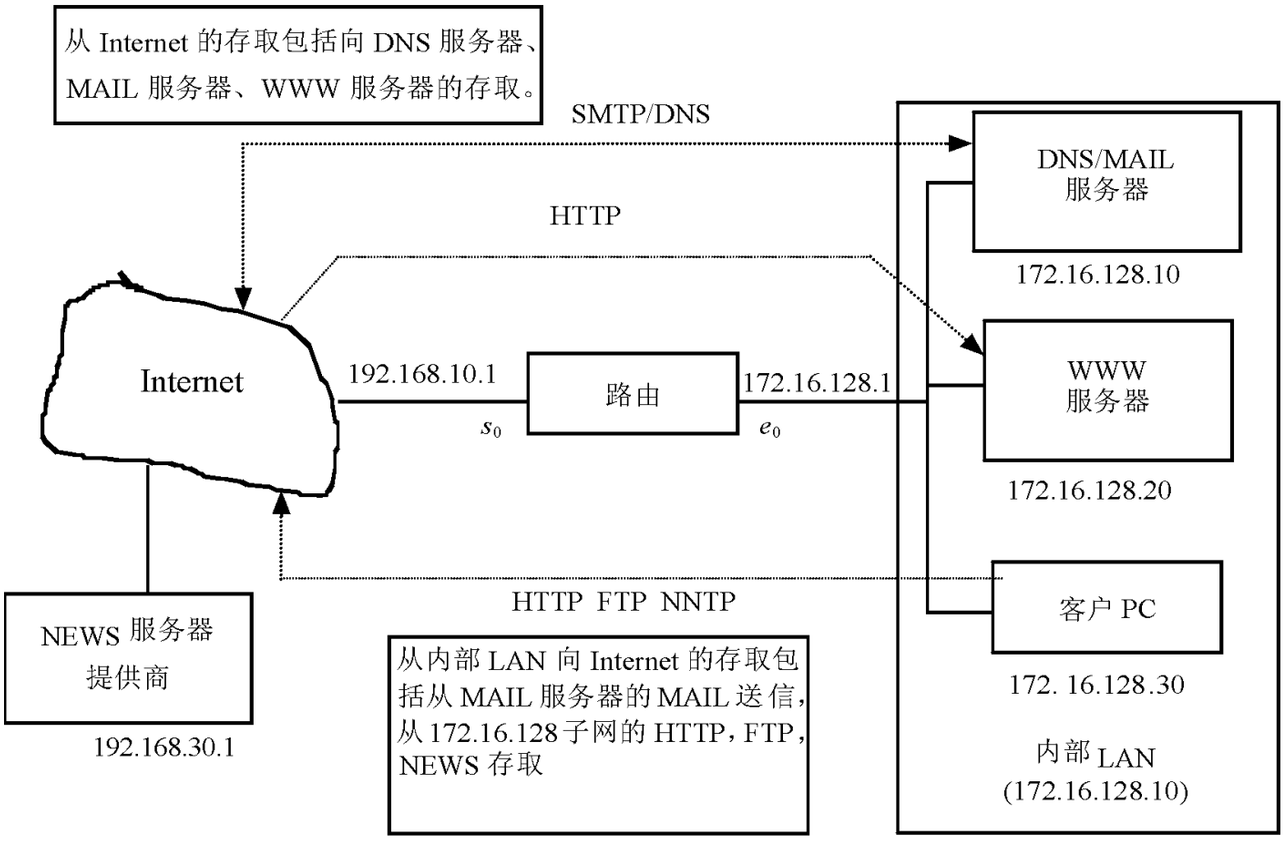


图 6 .14　基于 IP 分组过滤的 Internet 存取

序号	动作	协议	源地址	源接口	目的地址	目的接口	说　　明
1	许可	UDP	172 .16 .128 .10	53	任意	53	向 Internet 上的 DNS 服务器的查询
2	许可	TCP	172 .16 .128 .10	> 1023	任意	53	向 Internet 上的 DNS 服务器的查询
3	许可	TCP	172 .16 .128 .10	> 1023	任意	25	从 Mail 服务器向 Internet 上的 Mail 服务器送信
4	许可	TCP	172 .16 .128 .*	> 1023	任意	80	从局域网 LAN 向 Internet 上的 Mail 服务器送信
5	许可	TCP	172 .16 .128 .*	> 1023	192 .168 .30 .1	119	从 LAN 向 NEWS 服务器的存取
6	许可	TCP	172 .16 .128 .*	> 1023	任意	21	从 LAN 向 FTP 服务器的存取(通常的接口)
7	许可	TCP	172 .16 .128 .*	> 1023	任意	20	从 LAN 向 FTP 服务器的存取(被动模式数据)
8	许可	TCP	任意	20	172 .16 .128 .*	> 1023	从 LAN 向 FTP 服务器的存取(正常的模式数据)

使用防火墙能提高安全性。分组过滤和下面要讲的代理(proxy)组合在一起就可以

构成一个防火墙。

其中,DNS 为 domain name system,即域名系统,SMTP 为 simple mail transfer protocol,即简单邮件传送协议。

3) 基于代理服务的防火墙设计

建立防火墙的另一个途径就是基于代理服务的防火墙设计。基于代理服务的防火墙设计为内部网络用户提供了更好的安全性。代理系统通常包括两部分:代理服务程序和客户程序。代理服务程序在客户程序和真正的服务器程序之间起到一个中间结点的作用。客户程序与这个中间结点(即代理)连接,然后中间结点与真正的服务器进行连接。内外网络之间不存在直接连接。

所谓代理,指的是进行存取控制和过滤的程序,是位于客户机与服务器之间的中继。下面以 TIS 防火墙工具为例,来说明代理的基本功能。

(1) TIS 防火墙 Toolkit

TIS 防火墙工具可以从可信任的信息系统(trusted information system, TIS)公司的网页下载可能的源程序,源程序提供了防火墙构成软件。网页地址为: http:// www .tis .com/ docs/ products/ fwtk/ 。TIS-fwtk 包括存取控制程序、ftp-telnet 和 rlogin(远程注册)等代理、认证服务器软件。这可以用来有效地解释应用型网关的防火墙的基本功能。TIS 公司提供了防火墙制品,这成为 TIS-fwtk 的基础。

(2) plug-gw 的设定

TIS-fwtk 中包含的 plug-gw 是一个通用的代理,在网络新闻 NetNews 使用的 NNTP 传送协议(net news transfer protocol) (TCP 接口为 119)中进行中继设定,图 6 .15 提供了中继功能。

4) 虚拟私人网(VPN)

在 Internet 与企业网 Intranet 之间接续防火墙,进行存取控制提高了安全性。但是,防火墙不能防止对网上的信息盗听、篡改的攻击。为了保证信息不受攻击,数据加密是有效的方法。最近,在 Internet 上接续了具有加密功能的路由和防火墙,把网络上的数据进行加密再传送,这种方法叫做虚拟私人网功能。

在防火墙间进行加密通信使网络私设网化形成拟私人网的防火墙制品也出现了。带有虚拟私人网的防火墙的构成和功能如图 6 .16 所示。

许多防火墙有一些种类的 VPNS:加密的防火墙到防火墙通道。在一个防火墙和另一个防火墙之间所有的通信都是加密的。在远程现场,嵌入另一个网络协议软件包中,并且通过 Internet 发送,防火墙从网络协议软件包中提取加密的信息,并且解密它以得到原来的网络软件包。VPN 是必要的,因为在使用公开的网络(像 Internet)的双方之间通信是极易受偷听攻击的。所发生的危险取决于所传送信息的重要性。VPN 在关键路由或通信结点允许操作保证所有的网络通信都是安全的。Internet 的网络通信极易受到电子探听。

如果现场 A,B 之间设置了 VPN,那么在那些现场之间的通信都是加密的。VPN 还能代表在私人网络中通信成本的节省。1996 年 3 月美国计算机界一份报告指出连接 LAN 和 WAN 的 Internet 中使用加密的通信降低了成本 23%～50%。

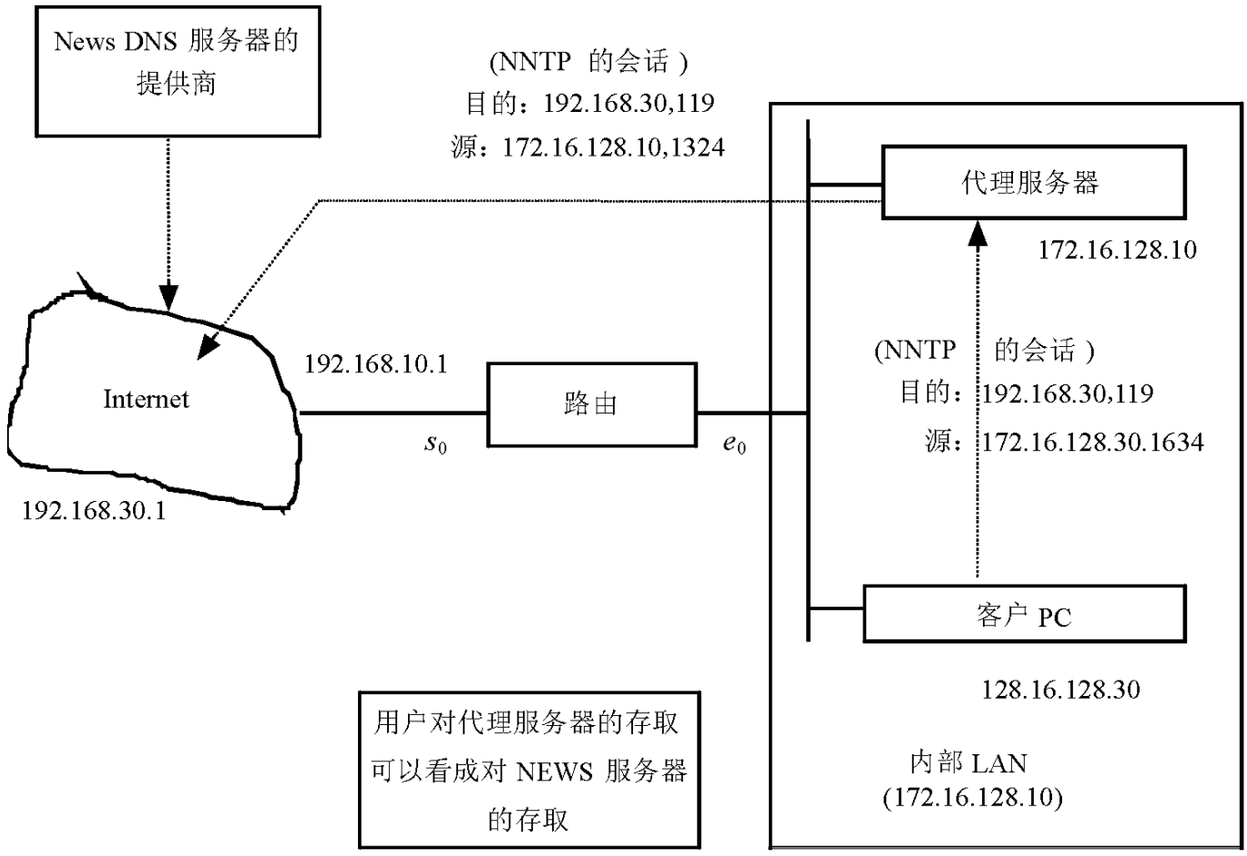


图 6.15 NNTP 中继(代理服务)

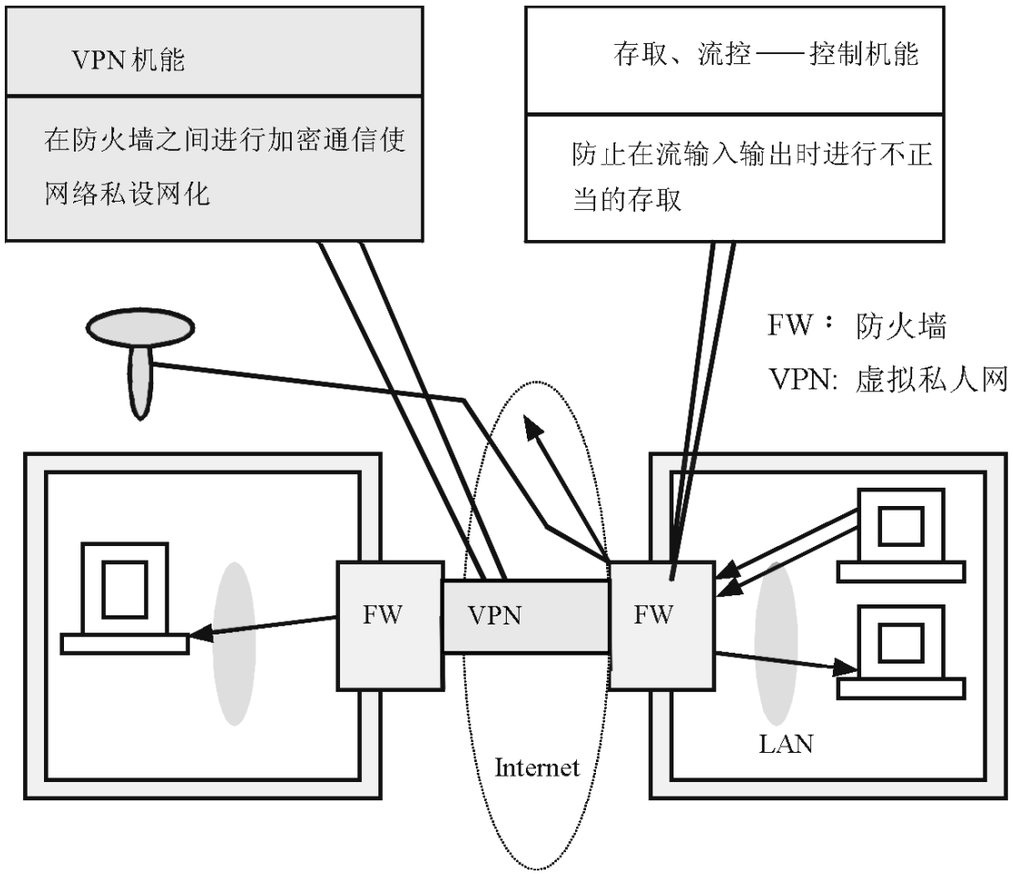


图 6.16 带有虚拟私人网的防火墙的构成和功能

因为在 Internet 上 VPN 变得广为使用，VPN 建立更自动化，所以大部分 VPN 用于没有完全可信关系的现场之间通信以使通信安全。所谓完全可信关系，我们指的是商业

伙伴、生产者与消费者、供应者与顾客之间的关系。通信私人网安全化是令人鼓舞的,一个Internet 防火墙可以用于控制和阻止对内部和私人网的存取。

6.4 利用 IP 欺骗进行攻击及其预防策略

本节首先介绍黑客是如何进行 IP 欺骗攻击的,然后介绍 IP 欺骗预防策略。

1. 利用 IP 欺骗进行攻击

利用 IP 欺骗攻击就是黑客盗用 IP 地址进行信息偷盗、篡改。在 Internet 领域,IP 欺骗已成为黑客攻击的一种主要的手段。黑客伪造 LAN 主机的 IP 地址,根据这个伪造的地址进行不正当的存取(图 6 .17)。1994 年 12 月发生的 IP 欺骗攻击的犯人 Kevin Mitnick 使用了一些巧妙的技术进行了 IP 欺骗。

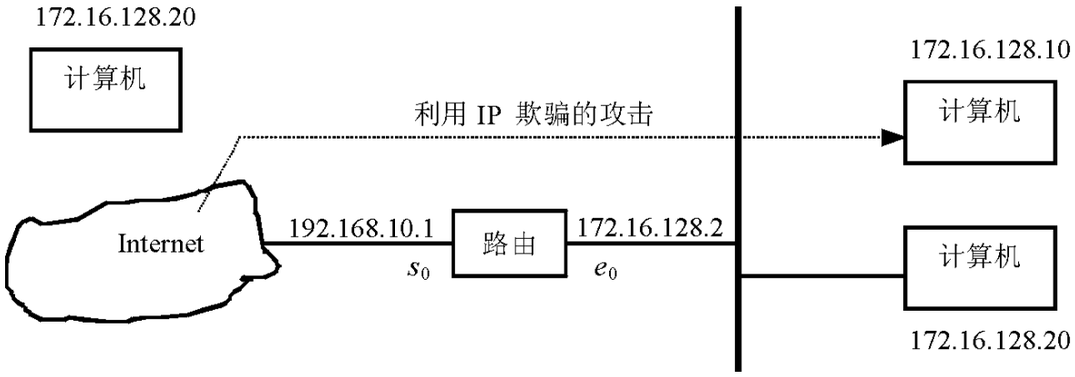


图 6 .17 利用 IP 欺骗的攻击

黑客为了进行 IP 欺骗,需进行以下工作:使得被信任的主机丧失工作能力,同时采用目标主机发出的 TCP 序列号,猜测出它的数据序列号。然后,伪装成被信任的主机,同时建立起与目标主机基于地址验证的应用连接。如果成功,黑客可以使用一种简单的命令放置一个系统后门,以进行非授权操作。黑客攻击的全过程如表 6 .8 所示。

表 6 .8 黑客攻击的全过程

时间序列	主机 A(目标主机)	控制(显示有关 TCP 控制字段头部的控制字符和该字段的流动方向)	主机 B(对于 A 来说,可信任的主机)
1	Z(B)	---SYN---->	A
2	B	<---SYN/ ACK----	A
3	Z(B)	---ACK--->	A
4	Z(B)	---PSH--->	A

主机 A 和主机 B 为参与一次 TCP 对话的机器,SYN 为数据序列,ACK 为数据确认,PSH 为压入功能(push function),Z 为进攻主机。

攻击者伪装成被信任主机的 IP 地址,此时,该主机仍然处在停顿状态(丧失工作能力),然后向目标主机的 513 端口(远程注册 rlogin 的端口号)发送连接请求,如时刻 1 所

示。在时刻 2,目标主机对连接请求作出反应,发送 SYN/ ACK 数据包给被信任主机。按照计划,被信任主机会抛弃该 SYN/ ACK 数据包。然后,在时刻 3,攻击者向目标主机发送 ACK 数据包,该 ACK 使用前面估计的序列号加 1。如果攻击者估计正确,则目标主机接收该 ACK。至此,连接正式建立起来了。在时刻 4,将进行数据传输。一般地,攻击者将在系统中放置一个门,以便侵入。经常会使用"cat + + > > ~/ .rhosts",这个办法迅速而简便地为下一次入侵铺平了道路。

2. IP 欺骗的预防策略

利用 IP 欺骗攻击比较普遍,而且产生的危害很大,所以 IP 欺骗预防很重要。

1) 抛弃基于地址的信任策略

阻止这类攻击的一种非常容易的办法就是放弃以地址为基础的验证。不允许 r* 类远程调用命令的使用;删除 .rhost 文件;清空/ etc/ hosts .eauiv 文件。这将迫使所有用户使用其他远程通信手段,如 Telnet,SSH 等。

2) 使用随机化的初始序列号

黑客攻击得以实现的一个很重要的因素就是,序列号不是随机选择或随机增加的。Bellovin 学者描述了一种弥补 TCP 不足的方法,就是分割序列号空间。每一个连接将有自己独立的序列号空间。序列号将仍然按照以前的方式增加,但是在这些序列号空间中没有明显的关系。可以通过下列公式来说明:

ISN = M + F(localhost, localport, remotehost, remoteport)

其中,M 为 4 微秒定时器,F 为加密 HASH 函数,localhost 为局部主机,localport 为局部接口,remotehost 为远程主机,remoteport 为远程接口。

F 产生的序列号,对于外部来说是不应该能够计算出或被猜测出的。Bellovin 建议 F 是一个结合连接标识符和特殊矢量(随机数,基于启动试卷的密码)的 HASH 函数。

3) 分组过滤

阻止 IP 欺骗的另一种明显的方法是进行分组过滤。对于通过路由器接入 Internet 的网络,可以利用自己的路由器来进行过滤,确信只有自己的内部 LAN 可以使用信任关系,而内部 LAN 上的主机对于 LAN 以外的主机要慎重处理。自己的路由器可以帮助自己过滤掉所有来自于外部而希望与内部建立连接的请求。

4) 使用加密方法

当有多种手段并存时,可能加密方法更为适合。这种方法是将要通信传输的信息进行加密,使其变成密文,然后再进行传输和验证。

6.5 面向对象分布式环境的认证与加密系统

由于 Internet 具有信息发送、共享和分布的优势,因此基于 Internet 获取信息的软件非常引人注目。在广域网环境中,很难进行软件的共享、分布和再利用。面向对象的分布式网络环境 OZ 项目就是为克服这个缺点而开发的一种基于 Internet 面向对象的分布式软件共享系统。面向对象的分布式网络环境的特点是在全球网上实现信息、数据、软件、

程序及服务(由服务器提供的特征)的分布、共享和传递,提高软件生产率和可靠性。使用面向对象的技术和密码学技术,能进一步地实现网络上信息的安全共享和交换。面向对象的策略是在网络上实现信息共享、分布和软件再利用的最合适的策略。

该项目是日本电子技术综合研究所开发的,属于开放型创造性基础软件项目,在 Internet 网上供全球使用,其中通信安全作为一项重要的技术指标而考虑。下面分别介绍一下研究的主要成果。

1. 认证系统

对于网络认证服务,美国麻省理工学院设计了一种 Kerberos(version 5 .0)系统,该系统可以对网络提供很好的认证服务,但是,该系统不一定能适合于所有的各种网络环境。这里有必要针对 OZ 的通信安全要求,具体问题具体分析,设计一个公开钥安全服务器。在 OZ 中,提供认证安全服务,不仅可以避免不正确的存取、更改、盗窃、中途攻击,而且能有效地管理密钥。

认证服务器是运行在一个安全机器上的一个程序,它通过发出一个需求、给出一个响应这样一个过程在网络上进行认证。caller 有个远程 callee 需要存取 caller 的私人网(caller, callee 可以是远程商业伙伴、远程软件销售商、远程客户、远程服务器、远程工程技术人员、远程会员等),那么 caller 需要明确地确认 callee 的身份,确认 callee 确实是要求存取时声明的真正的 callee,而不是伪造的。caller 和 callee 可以相互识别,caller 能识别 callee 的身份,同样 callee 也能识别 caller 的身份,辨别真伪。保证面向对象的分布式网络环境移动对象通信安全。

本节针对面向对象的分布式环境 OZ 的通信安全要求,应用密码学技术,设计了基于公开钥算法的安全认证系统,进行 caller 与 callee 之间的身份认证。采用面向对象(oriented-object)策略、使用最先进的 Java 语言实现了这个认证子系统,采用 JDK1 .1 .2 比采用其他程序设计语言更先进、方便,避免许多重复编程。

1) 认证模型

在大多数情况下,一个对象(全局对象和局部对象用 cell 表示)有两种处理方法:管理 cell 的方法和提供服务的方法。前者只由它的拥有者和管理者使用,而后者可能被不知姓名的用户使用。为满足这些要求,认证是必要的。

在远程方法调用开始,在不同的执行机上的 cell 之间,由执行机的通信机制进行认证。认证避免了非法攻击。将网络上传输的数据加密避免偷盗篡改。

在同一个执行机上的 cell 之间的远程方法调用,不需要进行认证。

图 6 .18 显示了实现 Needham-Shroeder 算法的,类似于 Kerberos 的密钥配布中心的安全服务认证模型。

认证过程处理如下:

caller 向公开钥安全服务器(public key security server,简称为 PubKSS)发送一个需求信息,需要一个“凭证”(credentials),PubKSS 响应这个信息,发行一个“凭证”,并用 caller 的公开钥将“凭证”加密。“凭证”由会话钥和通行票组成。caller 传送这个通行票给 callee, caller 和 callee 可以相互认证彼此的身份。

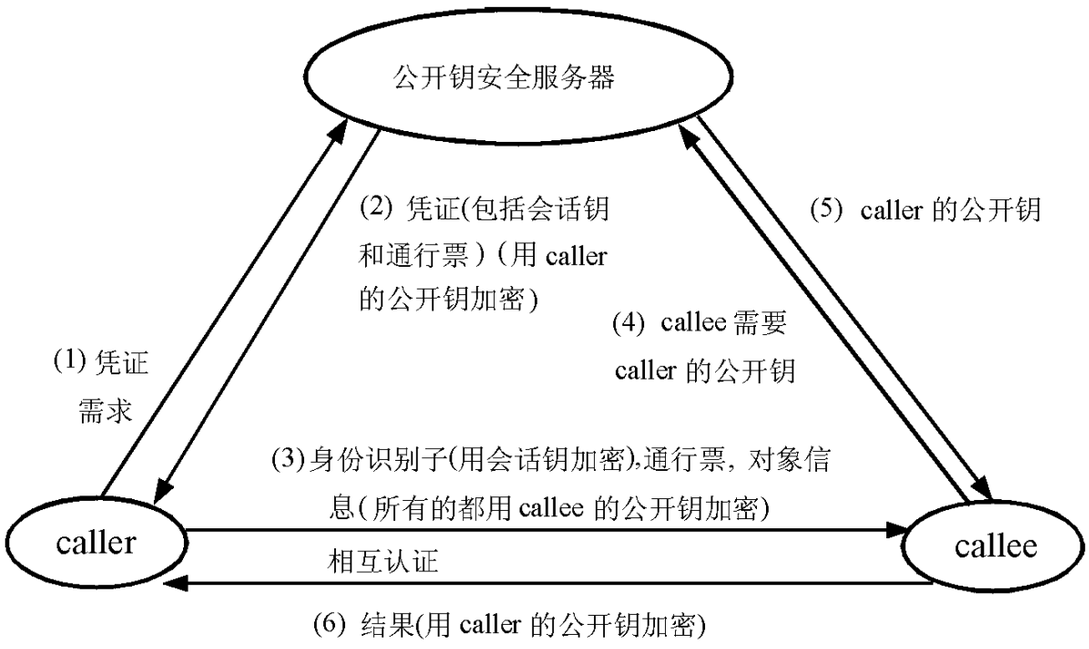


图 6 .18　认证模型

2) **认证服务信息交换和协议**

认证服务信息交换由两部分组成: caller **与** PubKSS **之间的认证服务信息交换,如图** 6 .19 **所示。**caller **与** callee **之间的认证服务信息交换,如图** 6 .20 **所示。**

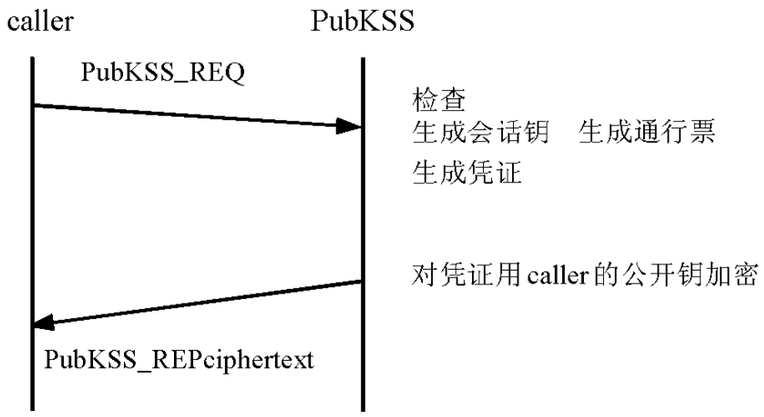


图 6 .19　caller 与 PubKSS 之间的认证服务信息交换

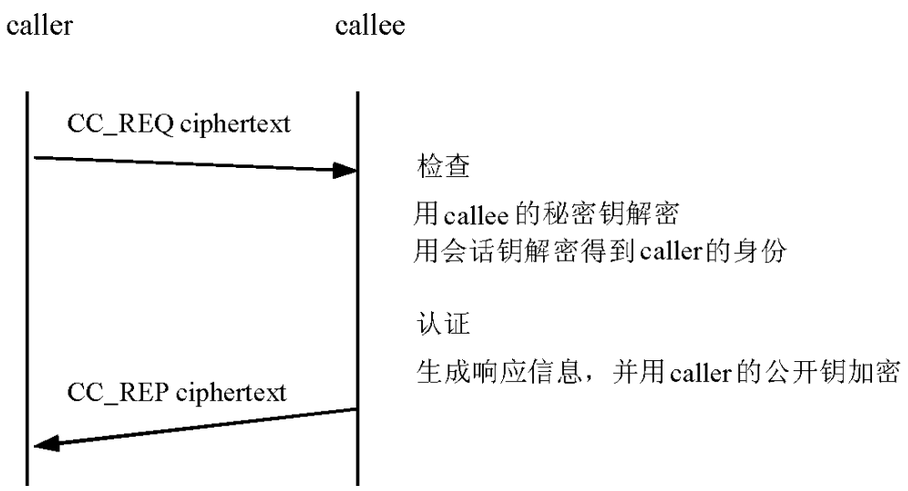


图 6 .20　caller 与 callee 之间的认证服务信息交换

(1) 公开钥安全服务器信息交换和协议

PubKSS_REQ 信息是“凭证”需求信息,

PubKSS_REQ = {callername, calleename, need, till_time }

callername, calleename 分别是 caller,callee 的名称,代表各自的身份。

need 表示 caller 需要 callee 的公开钥。

till_time 是通行票的有效期截止的时间,也就是说通行票过了这个时间,就会变得无效。有效期限过期的场合,需要从安全服务器再次取得凭证。

caller 准备好 PubKSS_REQ 后将它发送给公开钥安全服务器。

公开钥安全服务器接收到 PubKSS_REQ 之后作出响应。PubKSS 在它的数据库中查找 PubKSS_REQ 中定义的主体名 callername,calleename,如果他们是已注册的用户,则获取他们的公开钥。如果他们是未注册的用户,则给出错误信息。

公开钥安全服务器生成会话钥(sessionkey),生成通行票(ticket) .

sessionkey = {随机数}
ticket = { sessionkey , callername, calleename, authorithedtime, tickettill_time, publickeyofcallee}

callername, calleename 分别是 PubKSS_REQ 中的 callername, calleename。

authorithedtime 是公开钥安全服务器批准发行“凭证”的时间,用当前的系统时间表示。

tickettill_time 是批准的有效期截止的时间,

如果 PubKSS_REQ 中要求的通行票的有效期截止时间 till_time 为 0,那么批准的有效期截止的时间 tickettill_time 是当前的系统时间加上 1 天,否则批准的有效期截止的时间 tickettill_time 是 PubKSS_REQ 中的 till_time。

形式化描述如下:

if till_time = 0 in PubKSS_REQ
then tickettill_time = 当前的系统时间 + 1 天
else tickettill_time = till_time in PubKSS_REQ

publickeyofcallee 是 callee 的公开钥。

“凭证”credentials = {sessionkey, ticket}

将 credentials 用 caller 的公开钥 publickeyofcaller 加密形成密文 credentialsciphertext

credentialsciphertext = RSA_ENCODE(credentials, publickeyofcaller)

其中 RSA_ENCODE 是使用 RSA 公开钥算法的加密变换。

公开钥安全服务器将 credentialsciphertext 发送给 caller。

caller 接到 credentialsciphertext 后用自己的秘密钥 privatekeyofcaller 将 credentialsciphertext 解密,而后得到 sessionkey, ticket。

RSA_DECODE(credentialsciphertext, privatekeyofcaller) credentials

其中 RSA_DECODE 是使用 RSA 公开钥算法的解密变换。

(2) caller/ callee(CC)认证服务信息交换和协议

caller 准备 CC. REQ 信息之后发送给 callee 一方。

CC. REQ = {callername, authenticator, ticket, callerobjectmessage}

其中 authenticator = {callername, callertimestamp}

authenticator 称作 caller 的身份识别子。callertimestamp 是 caller 的时间邮戳。

callerobjectmessage 是 caller 发往 callee 的对象信息。

利用 DES 算法,使用 sessionkey 作密钥将 authenticator 加密形成 acipher,

acipher = DES. ENCODE(authenticator, sessionkey)

其中 DES. ENCODE 是使用 DES 对称钥算法的加密变换。

将 authenticator 加密后的 CC. REQ 表示为 CC. REQ1,那么

CC. REQ1 = {callername, acipher, ticket, callerobjectmessage}

CC. REQ1 是 caller 要发送给 callee 的信息,为安全传递,所以需要加密。这时采用 RSA 算法使用 callee 的公开钥 publickeyofcallee 对 CC. REQ1 进行加密形成 CC. REQciphertext。

CC. REQciphertext = RSA. ENCODE(CC. REQ1, publickeyofcallee)

caller 将 CC. REQciphertext 发送给 callee,以便 callee 认证 caller 的身份。

callee 接收到 CC. REQciphertext 后,首先用自己的秘密钥解密,以获得 CC. REQ1,从 CC. REQ1 中得知通行票 ticket。

RSA. DECODE(CC. REQciphertext, privatekeyofcallee) CC. REQ1

从通行票 ticket 中能得知会话钥 sessionkey,用 sessionkey 将身份识别子的密文解密以得到身份识别子明文信息 authenticator。

DES. DECODE(CC. REQ1, sessionkey) authenticator

其中 DES. DECODE 是使用 DES 对称钥算法的解密变换。

callee 取系统时间表示为 systemtime,假设偏差时间为 5 秒。如果系统时间 systemtime 比发行证明书的时间小 5 秒,那么说明通行票还没有生效。如果批准的有效期截止时间超过系统时间 5 秒,那么说明通行票已经过期,重新生成通行票。

形式化描述如下:

if systemtime - authorithedtime > 5 秒
then 通行票还没有生效;
if tickettill. time - systemtime > 5 秒
then 通行票已经过期,重新生成通行票 ;

如果解密的用户身份匹配 caller 的身份,并且证实数据是最新的,那么成功地完成了对 caller 身份的认证,否则 caller 是伪造的。

callee 需要 caller 的公开钥,发送一个信息给公开钥安全服务器,取回 caller 的公开钥

publickeyofcaller。

callee 将发送结果信息 result - REP 给 caller，result - REP 构成如下:

result - REP = { authenticator, resultobjectmessage }

用 caller 的公开钥将 result - REP 加密变成 result - REPciphertext，而后传送给 caller。

result - REPciphertext = RSA - ENCODE(result - REP, publickeyofcaller)

当 caller 收到 result - REPciphertext 后,首先用自己的秘密钥将 result - REPciphertext 解密 RSA - DECODE(result - REPciphertext, privatekeyofcaller) result - REP

将 result - REP 中的 authenticator 与自己原有的 authenticator 比较,如果符合,那么说明 callee 是可信赖的,不是伪造的,否则 callee 是伪造的。

认证过程由执行机自动实现。

2. 加密系统

针对面向对象的分布式环境 OZ 的通信安全要求,具体问题具体分析,提出了加密通信对象模型,设计了一种新的加密算法,充分采用面向对象的策略,使用最新的 Java 语言来实现这个加密系统,为面向对象的分布式环境 OZ 通信提供安全保障。由于在分布式面向对象的环境中网络上的通信被认为主要是对象之间的流通信,所以所设计的这个加密系统应该适合网络中的流通信。这个系统不仅适合于分布式面向对象的环境,也适合于像声音、图象这样连续介质的加密。

1) 加密通信对象模型

在面向对象的分布式环境(OZ)中启动远程方法,全局对象间通过 Internet 能进行数据通信。在 Internet 网上,重要的通信数据可能被盗听或修改,针对这种情况,所采取的对策就是将通信内容加密后再送信。

通信双方首先进行交涉,即交流、传递加密方式、加密、解密密钥等协议信息。交涉以全局对象为单位,进行协议交流。

(1) 送信处理

远程全局方法启动开始时,选择加密器,发送双方交涉协议。根据交涉协议结果,将要发送的明文信息流进行加密后再送信。

(2) 受信处理

接受发送来的密文信息流,根据交涉协议,选择解密器及密钥,将密文解密以获得明文信息流。

本节提出一种加密通信对象(object)模型,如图 6.21 所示。该模型确保网络中的流通信安全及安全方法调用。caller 要发送对象信息给 callee,安全的策略就是将信息加密之后通过网络传送,callee 接收到密文信息后,先解密,就会知道明文信息。callee 发送给 caller 对象信息也采用同样的安全策略。

在协议交涉与对象传递的过程中采用双往复通信模式,如图 6.22 所示。

2) 加密系统的算法

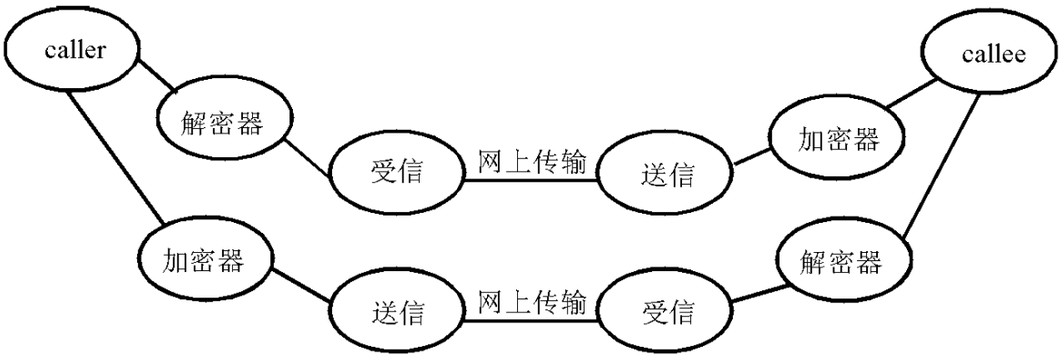


图 6.21　加密通信对象(object)图

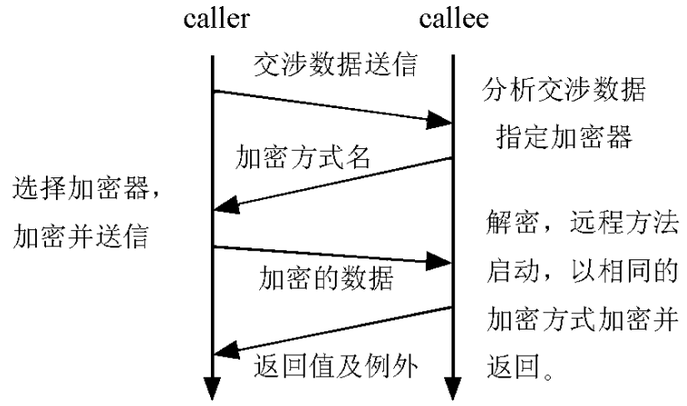


图 6.22　双往复通信模式

在面向对象的分布式环境 OZ 的流信息加密研究中，根据不同的保密强度要求，主要采用三种算法。

(1) 基于有限域的信息加密算法

基于有限域(finite field)的信息加密算法，简称为 FF 算法。FF 算法描述如下。

构造一个有限域：

$$V = \{0, 1, \cdots, 256\}$$

$$w = |V| = 257$$

令 X 是明文信息码集，Y 是密文信息码集

$$X = \{0, 1, \cdots, 256\}$$

$$Y = \{0, 1, \cdots, 256\}$$

所设计的基于有限域的加密算法是从 X 到 Y 的一个一一映射。

加密 $JM: X \to Y$

$\forall x \in X$，存在唯一的 y，并且 $y \in Y$

$JM: x \mapsto y$

而解密 $JM^{-}: Y \to X$

$JM^{-}: y \mapsto x$

$$\forall x_1, x_2 \in X, x_1 \neq x_2 \quad JM(x_1) \neq JM(x_2)$$

在加密算法中，

$$\forall y \in Y, \exists x \in X, JM(x) = y$$

一对一映射是一种满射，这保证了加密解密过程中明文与密文的一一对应。选择合

适的分块尺寸,该算法比 DES 算法保密强度高,并且运算速度也比 DES 的速度快。适合大量信息的加密。

(2) DES 算法

DES 是美国国家标准局宣布的商用数据加密标准算法,它是一个基于 64bits 并且进行 16 圈变换的算法,被广泛的使用。

(3) RSA 算法

在保密强度要求很强的情况下,利用公开钥 RSA 算法进行加密,其中 P 采用 512bits,或 768bits 或 1024bits。另外,采用 RSA 方法来保护对称密钥。RSA 算法保密强度是基于大数分解的难度。

对应以上三种算法设计了三个加密器,加密模型如图 6 .23 所示。

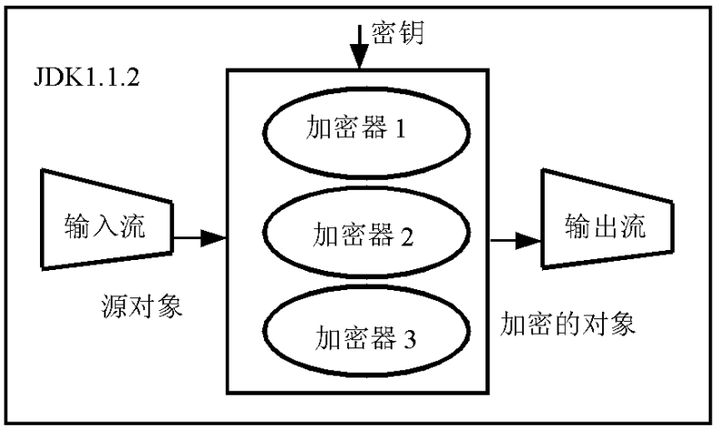


图 6 .23　加密模型

3) 类模型设计

该加密系统采用面向对象的策略进行设计和实现。在这里,类被认为是软件资源的单位。类有高度的模块性并支持多重继承。或者把已存在的类作为当前类的扩展,或者作为其他软件的一部分。类管理作为一个对象而实现。设计不同的类,利用类的继承、方法重载、模块封装等特点实现软件共享和再利用。设计了多个类,其中主要的类 Class OZCipher 描述如下。

它具有加密、解密功能,是一个抽象的类。

```
public abstract class OzCipher {
    String CipherName;
    protected OzCipher(String CipherName) throws Exception {
      this . CipherName = CipherName;
    }
    public abstract void encode(InputStream in, OZKey key, OutputStream out) throws Exception ;
    public abstract void decode(InputStream in,OZKey key, OutputStream out) throws Exception ;
    public String getName( ) throws Exception {
      return CipherName;
    }
}
```

其中 CipherName 是加密方式名称,如 FF,DES,RSA。in 是输入流,out 是输出流,key 是 OZKey,OZKey 是一个密钥对象。

OZKey 类描述如下:

```
package JP.go.ipa.oz.system;
import java.io.*;
/* 这是 OzKey 的源程序 */
public abstract
class OzKey {
   String Key = new String();
   OzKey(String key) {
      Key = key;
}
/* 返回密钥 */
String getKey() {
      return Key;
      }
}
```

对应三个算法设计三个加密器类 OZFFCipher, OZDESCipher, OZRSACipher, 它们继承 OZCipher, 具体实现 OZCipher 中的两个抽象的方法 encode(), decode()。继承关系模型如图 6.24 所示。在 RSA 方法的密钥生成及加密解密实现过程中充分利用 Java.math.BigInteger 避免重复编程。Java.math.BigInteger 提供了任意精度的大数的模代数运算, 如最大公约数计算 gcd(BigInteger), 素数生成测试 BigInteger(int bitLength, int certainty, Random rnd), 余数计算, 指数运算 modPow(BigInteger), 乘法逆元素计算 modInverse(BigInteger)。使用这些方法进行程序设计是很方便的, 这就是使用 Java 的优点之一。如果使用其他程序设计语言, 这些都要自己编程。

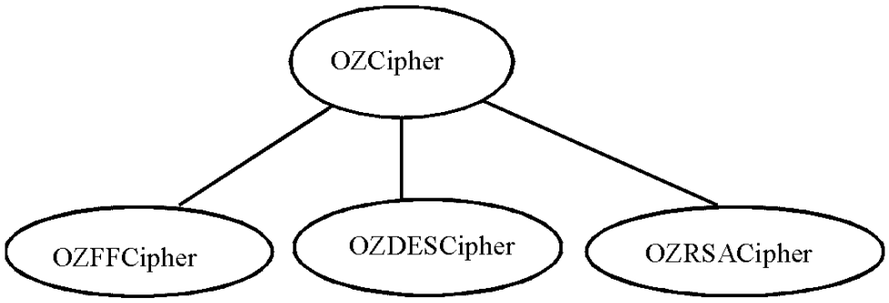


图 6.24 加密器类继承模型

4) 密钥交换协议

加密器列表列出了三种加密器名称, 通信双方可根据需要协商确定采用哪一种加密算法。如果采用对称密钥算法, 就需要一个可信任方法来分配传递共享密钥。然而在 Internet上没有安全通道交换秘密密钥。采用 RSA 方法来保护对称密钥。一般情况下, 加密算法是公开的, 秘密全部寓于密钥之中, 密钥的管理极其重要。

(1) 密钥的产生

· 基于有限域的算法 FF 及 DES 算法密钥的产生: 利用随机数产生器随机生成 0-1 比特序列。

· RSA 密钥的产生

按照 RSA 算法实现的步骤, 利用 Java.math.BigInteger 中现成的方法 BigInteger(int

bitLength, int certainty, Random rnd)等,计算密钥。

(2) 密钥的分配

一级文件密钥(数据密钥)由二级密钥(加密一级密钥的密钥)加密后,在双方交涉送信时传递。对一级密钥的加密采用 RSA 算法。

(3) 对二级密钥的保护

采用阈值方案保护二级密钥(D):将 D 分成 w 个影子 $D_1, D_2, \cdots, D_w$,使

· 知道任意 m 个或更多的 D_i ($1 \leq m \leq w$),就能有效地计算出 D;

· 知道任意 $m-1$ 个或更少的 D_i,由于信息不够而无法计算出 D;

w 个影子可以分给 w 个用户。由于重新构成密钥需要 m 个影子,所以只要暴露影子数不到 m 个,就不会危及密钥。任何少于 m 个有效的用户的组合都不能拼凑出密钥。同时即使有若干个影子丢失或损坏,只要还有 m 个有效的影子,仍可以恢复密钥。

5) 该加密系统的实现特点

本系统应用 JDK1.1.2(Java Development Kit 1.1.2)在 Sun Solary 工作站上开发实现的。Java 是面向对象的分布式的通用的基于类的网络程序设计环境。Java 具有程序的健壮性,使程序可靠,Java 也具有突出的安全性。Java 作为一种新技术被广泛的采用。加密系统的实现有如下特点。

(1) 内存管理安全

在 RSA 算法中,密钥 bits 长度很大(512,768,1024),计算复杂,要求安全有效的内存管理。其他面向对象的语言都要求程序员记录跟踪所产生的所有对象,并且当不再需要时销毁它们。用这种方式管理内存很烦琐,并且经常出错。我们采用 Java 的 Garbage Collection 技术,自动地进行内存管理,避免不能正确地管理内存的安全问题(如 C 中的 free 或 C++中的 delete)。Java 允许程序员创立所需要的尽可能多的对象,但从来不毁坏它们。当决定对象不再被使用了,Java 运行时,环境将删除这些对象。通过调用 System.gc()可随时运行 Garbage Collector,释放不再被需要的对象所使用的内存,采用 byte[]数组来存取大数,避免指针运算。

(2) 可供 Internet 网上共享和再利用

采用了面向对象机制,设计了多个类,设计了类之间的继承关系,充分重载类的方法。Java 获得巨大成功的一个关键因素是:程序编制一次,调试成功,可以在网上任何一个地方运行。采用 Java 语言,所编制的程序是 100%的纯 Java 程序,也具有这个特点:编写一次,调试成功,可以在网上任何地方运行,这样充分体现了面向对象的机制,发挥了 Java 的先进性。该加密系统不仅适合于 OZ 分布式网络共享环境,也适合于其他的分布式网络系统的加密。使该软件资源在 Internet 上被共享和再利用。这可以大量的节约软件开发的成本和时间,提高软件的生产能力。

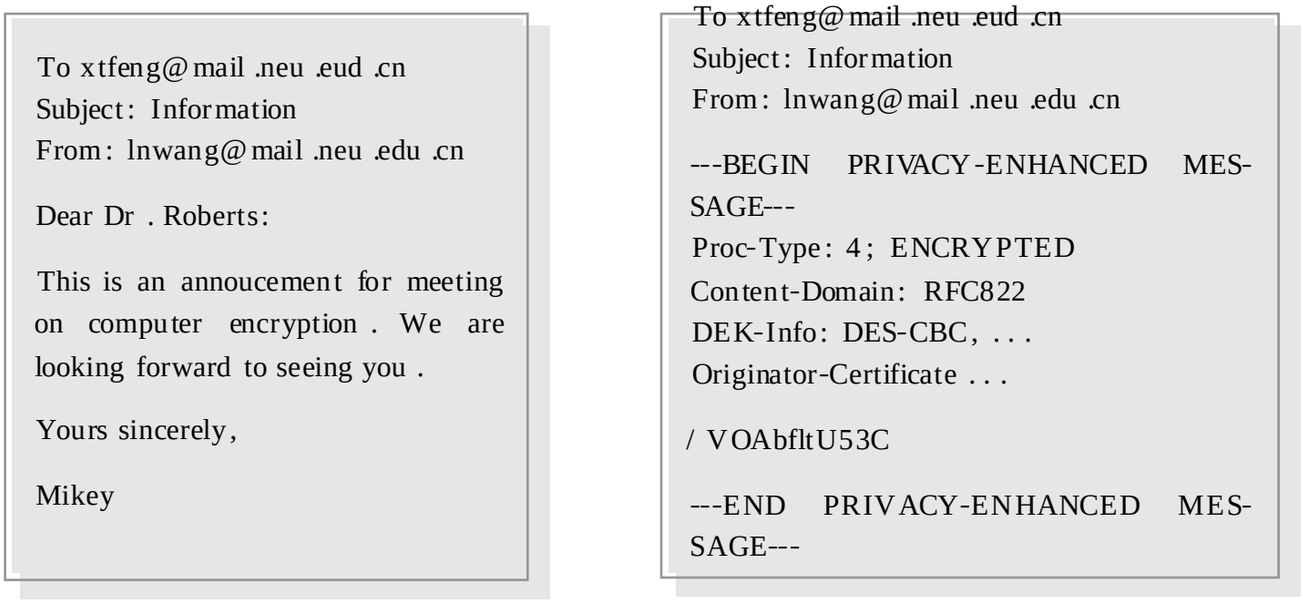
(3) 充分利用 Java API 资源,如在 RSA 算法实现的过程中,利用 math.BigInteger 中的 multiply(), modInverse(), gcd(), modPow()等现成的方法,免去重复编程。

6.6 秘密的电子邮件 PEM

秘密电子邮件 PEM(privacy enhanced mail),是面向 Internet 的秘密邮件标准,为在

Internet上传送 E-mail 提供安全保障。它从 1987 年开始研究,随着标准化作业的推进,逐渐形成了现在的标准状态,并以 RFC1421-1424 规格标准而发表。PEM 协议提供加密、验证、信息完整性保护和密钥管理功能。

PEM 信息与一般的 E-mail 信息的不同之处在于 PEM 信息是密文,且密文的前头有“---BEGIN PRIVACY-ENHANCED MESSAGE”,后头有“---END PRIVACY-ENHANCED MESSAGE”。图 6.25 给出了两种 E-mail 形式。



(a) Internet 上 E-mail 的一般形式　　(b) PEM 信息的形式

图 6.25　两种 E-mail 形式的比较

1. PEM 信息的形成

PEM 信息的形成包括:数据的正规化、数字签名的形成、数据加密(可选项 option)和码的可视化(局部化)。接收方正好经历一个逆过程,进行解密及验证签名。PEM 信息处理流程如图 6.26 所示。

1) 数据正规化

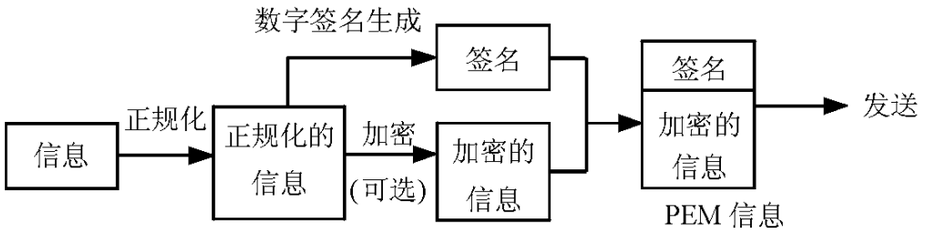
用户所使用的机器种类不同,使得文本数据表现形式也不同。所谓正规化就是把信息变换成为全部程序都能接收的共同的形式。例如,换行在 UNIX 系统中为 < LF > (ASCII Code 10),在 MS-DOS 中为 < CR > < LF > (ASCII Code 13 10),在 Macintosh 中为 < CR > 表示。由此在 UNIX 环境下作成的 PEM 信息,在 MS-DOS 上也能读,而不会因机种不同而不同。通用的数据正规化处理平台如图 6.27 所示。PEM 的正规化具体内容包括:文字码是 ASCII;行末与 MS-DOS 相同 < CR > < LF >。

2) 数字签名的形成

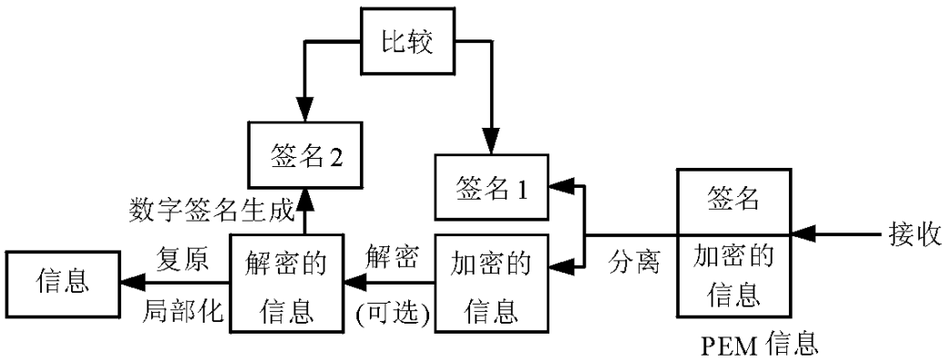
接下来介绍数字签名是怎样形成的。数字签名处理包括两步:信息、数字融合的抽取和信息、数字融合的加密。

(1) 信息数据的抽取

PEM 规定中使用的算法是 RSA-MD2 和 RSA-MD5 算法。其中 MD2, MD5 是由 Ron Rivest 专家设计的哈希函数,产生 128 比特值。PEM 中抽取的信息融合有一个信息完整



(a) 发信方的信息处理流程图



(b) 接收方的信息处理流程图

图 6 .26　PEM 信息处理流图

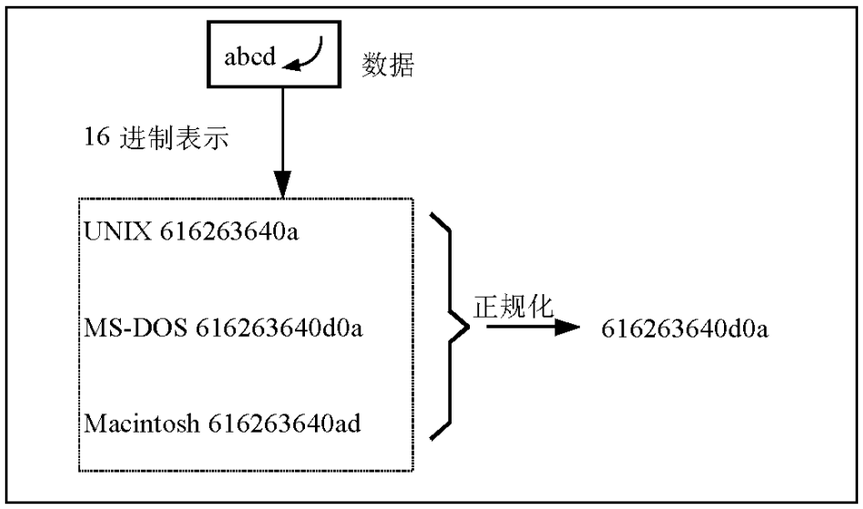


图 6 .27　数据的正规化处理

性检查 MIC(message integrity check)**的过程。**

(2) MIC **信息数据的加密**

为了在通信线路上使 MIC **不被改变,采取了加密处理措施。为此所使用的加密算法可以是对称钥加密和非对称钥加密方式。因为验证数字签名比较容易,所以考虑可以使用非对称钥方式对** MIC **加密。例如,采用** RSA **方法对** MIC **进行加密。数字签名是用送信者本人的** RSA **秘密钥对** MIC **进行解密后形成的,该数据在** MIC-info **域中进行了说明,并需要进行可视化处理。**

3) **数据的加密(可以选择)**

签名处理中有必要对数据进行保密的情况下,则对信息原文要进行加密处理(这一项任选项,可做可不做)。加密过程如图 6 .28 **所示。**

PEM **的加密处理步骤如下:**

(1) **数据加密钥** DEK(data encryption key)**的生成**

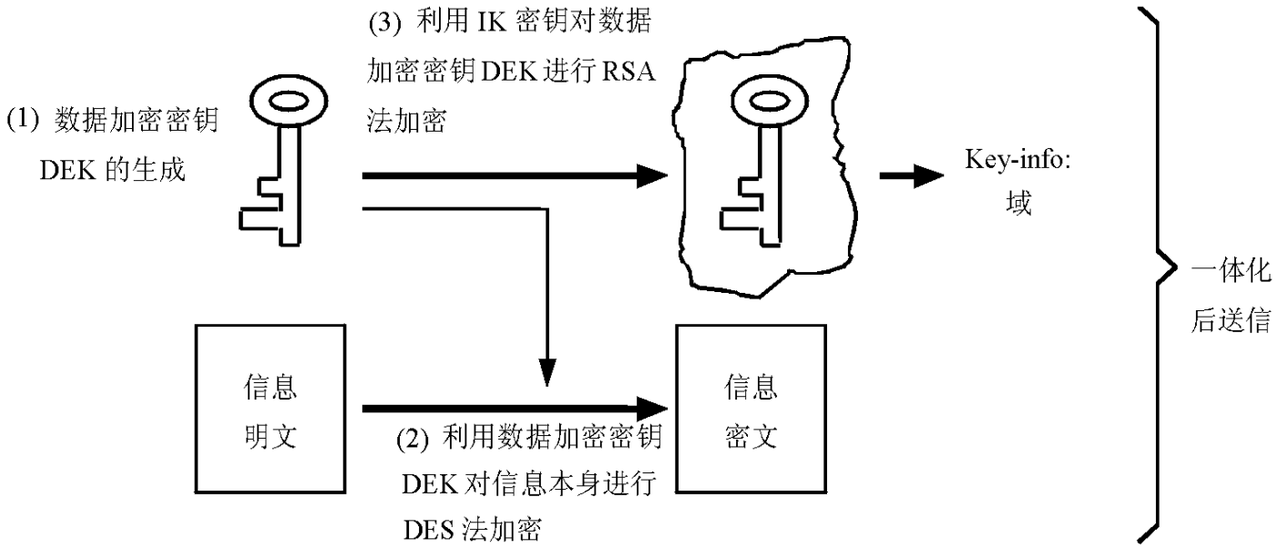


图 6 .28　PEM 信息加密处理过程

(2) **用 DEK 密钥将信息本体加密**

(3) **用交换密钥 IK(interchange key)把 DEK 加密**

密钥 DEK 是一组随机产生的数,采用 DES 算法使用 DEK 作密钥将信息本体加密,用 IK 作密钥使用 RSA 算法把 DEK 加密。

对于 DEK 的加密,一般使用非对称钥加密算法 RSA,并且用 IK 作密钥。IK 是受信者的公开钥。在受信者比较多的情况下,用各个受信者的 IK 对 DEK 加密,要能区分不同的受信者,所以有必要使用两个信息域: DEK-Info 和 Recipient-ID, 其中, DEK-Info 表示加密的 DEK 和加密算法名称, Recipient-ID 表示受信者名称, 如图 6 .29 所示。将 Recipient-ID 信息域配置在信息头部位置,保证在信息传递过程中免除顺序号被改变的危险性。

4) **码的可视化**

不管使用对称钥方式还是非对称钥方式,加密后的数据都是 8 比特整数倍的数据,在 Internet 上传送这样的信息是不合适的。为此把加密后的数据进行相应的处理,以使电子邮件变成可读的形式,这个过程称为码的可视化。对任意的 8 比特数据的可视化处理,著名的例子是 UNIX 的 uuencode。在 PEM 中,不是简单地使用这样已存在的方法过程,而是开发了新的处理方法。如图 6 .30 所示,表 6 .9 表示了码的可视化处理变换对应关系。

码的可视化处理描述如下。

(1) **将 16 进制的源数据转换为二进制表示,以 6 比特为单位分割成若干组。如十六进制数 F862943E0A5C(是 3 的倍数)可以转换为 8 组 6 个比特的二进制数 111110, 000110,001010,010100,001111,100000,101001,011100。**

(2) **把每组的二进制数转换成十进制数。例如(1)步中的 8 组二进制数转换成十进制数为 62,6,10,20,15,32,41,28。**

(3) **查表 6 .9,把十进制数映射成相应的字符,例如 + GKUPgpc。如果最后一组二进制不足 6 比特,则在低位部分补足 0。如果全体文字为 4 的倍数,则用' = '结尾。**

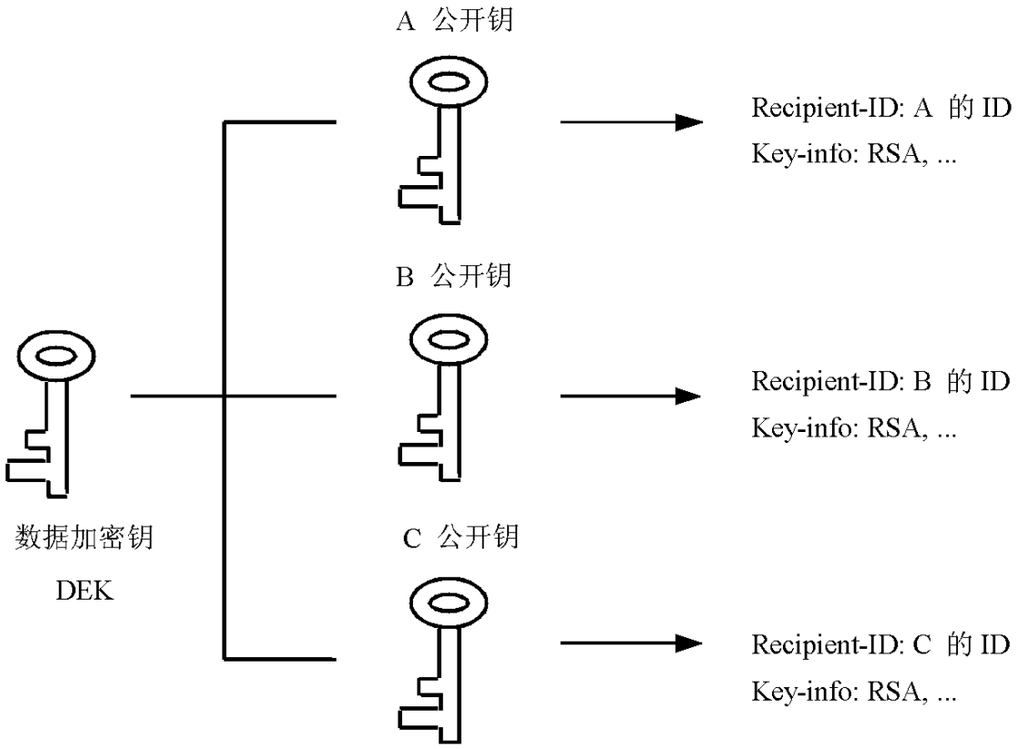
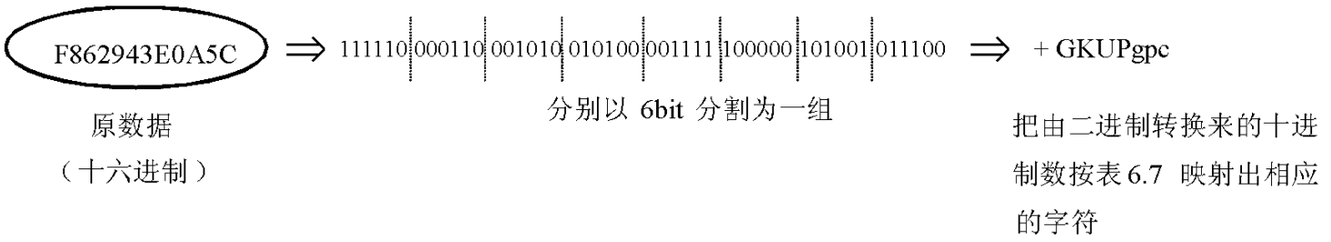
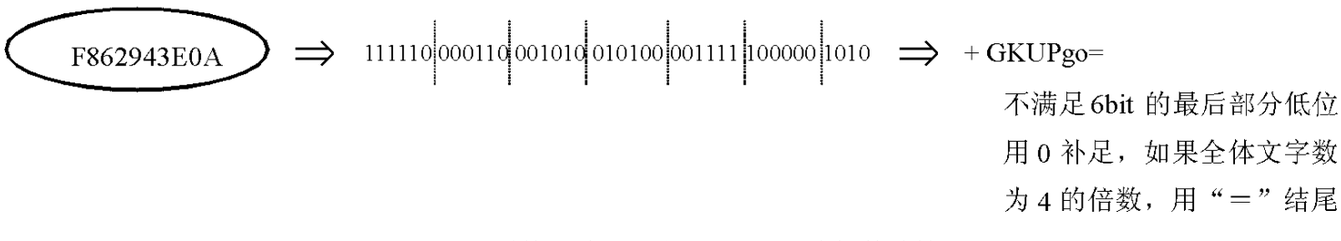


图 6.29　数据加密密钥 DEK 的加密处理方式



(a) 原数据的尺寸(byte)是3的倍数的情况



(b) 原数据的尺寸(byte)不是3的倍数的情况

图 6.30　码的可视化处理过程

表 6.9　码的可视化处理变换表

0	A	7	H	14	O	21	V	28	c	35	j	42	q	49	x	56	4	63	/
1	B	8	I	15	P	22	W	29	d	36	k	43	r	50	y	57	5	(pad) =	
2	C	9	J	16	Q	23	X	30	e	37	l	44	s	51	z	58	6		
3	D	10	K	17	R	24	Y	31	f	38	m	45	t	52	0	59	7		
4	E	11	L	18	S	25	Z	32	g	39	n	46	u	53	1	60	8		
5	F	12	M	19	T	26	a	33	h	40	o	47	v	54	2	61	9		

6	G	13	N	20	U	27	b	34	i	41	p	48	w	55	3	62	+

5) 信息全体的构成

这里介绍根据以上原理完成的信息构成。PEM 信息的类型包括：进行过签名但未进行过可视化处理的信息 MIC-CLEAR、进行过签名并进行过可视化处理的信息 MIC-ONLY、密文信息 ENCRYPTED MIC。

下面分别介绍 MIC-CLEAR 信息和 ENCRYPTED MIC 信息。

(1) MIC-CLEAR 信息

图 6 .31 表示了 MIC-CLEAR 信息的一个例子。

Pro-TYPE：这个域表示过程类型和 PEM 类型。

Content-Certification：这个域表示范围，如 RFC822。

```
To: xtfeng@mail.neu.edu.cn
Subject: Singed Message
Content-type: text/plain; Charset = ISO-2022
Data: Sat.1 August 1998 13:07:48 + 0900
From 王丽娜 = <lnwang@mail.neu.edu.cn>

---BEGIN PRIVACY-ENHANCED MESSAGE-----
Proc-Type:4; MIC-CLEAR
Content-Domain: RFC822
Originator-Certificate:
MIIBxzCCAWUCAg09MAOGCSqGSIb3DQEBAgUAMD4xCzAJBgNVBAYTAkpQMQOwCwYDVQQKEwRXSUF
AwHgYDVQQLExdDZXJOaWZpY2F0aW9uIFF1dGhvcmIOeTAeFw05NTA1MTYwNDMyMjJaFw05NzA1MTYw
yMjJaMIGRMQswCQYDVQQGEwYKUDEfMB0GA1UEChMWTkVDIEIuZm9ybWF0ZWMgU3IzdGVtczEjMCEG
ECxMaQWR2YW5jZWQgA1UEChMWTkVDIEIuZm9ybWF0ZWMgU3IzdGVtczEjMCEGA1UECxMaQWR2YW5
QgVGVjaG5vbG9neSBTeXN0ZW0xPDASBgNVBAMTC0IOQU1VUqEgWW91MCYGCSqGSIb3DQEJARYZaW5h
VyYUBhdHMubmIzLn51Yy5jby5qcDBcMA0GCSqGSIb3DQEBAQUAA0sAM1gCQQCjU9yiyHq3/ E-
BCYSXw0Y3itE
7ym +
dQgi2MLxCY550sr48F2wiqiLTemV0Tm2f4mE3hJIbI3YrkoI1IK9DngfUDAgMBAAEwDQYJKoZIhvcNAQBEC
BQADTQBZrksfauX9HEaS9rS4qn1goTJNHMFFgKxEF6Zx1N82VkXm9mqrh1iGJaflZoie2Z00QKqhYQCSc
+IIqY1C
fZCswfof3R0mQphZQjKW

Issuer certificate:
MIIBfjCCARwCATwwDQYJKoCIhvcNAQECBQAwPjELMAkJA1UEBhMCSIAxDTALBgNVBAoTBFdJREUxIDA
VBAsTF0NIcnRpZmIjYXRpb24gQXVOaG9yaXR5MB4XDTkzMTEwOTAwMDAwMFoXDTk2MTEwOTAwMDA
wPjELMAkGA1UEBhMCSIAxDTALBgNVBAoTBFdJREUxAIDAeBgNVBAsTFONIcnRpZmIjYXRpb24gQXY0aC
R5MGgwDQYJKoZIhvcNAQEBBQADVwAwVAJNAG9zTm8/ Lpo3bvxog72830dT0oSPL2v7Vf9bqQQ +
INVW5pZyT
QnyH8hEbTSCIn/
Vdjdph6j10k43WcPtSthSB4ggysVGXdHW3YQTAkCAwEEATANBgkqhkIG9w0BAQIFAANADuE
nfRO5r342YWhnYyn6ETxIU6dzd4JMbKgTzOrp0hQQtgF/ 9HxkhjE49NUP00VerZPKQUu7ppt2 + v/
ZFahDIOIhXQI3
J8gUWLj

MIC-info: RSA-MD5,RSA,
  IPntTUasN3zKK/ Bop9o2HEk8LubN7XePhTL59Yqm3kDvgIrzLMYPe7/ CVixRwISNzP9oLWynZ3X
od0iHikXAok = =
---END PRICACY-ENHANCED MESSAGE---
```

图 6.31 MIC-CLEAR 信息

Originator-Certificate: **这个域表示发行者的证明书**(Certificate)。

Issuer-Certificate: **这个域表示发行证明书的机关**。

MIC-Info: **这个域表示信息融合/ 加密的算法名称**。

(2) **加密后的** MIC **密文信息**

采用上面介绍的加密处理方法,对明文信息加密,获得了密文信息。图 6.32 **表示了加密后的信息。**

```
To: xtfeng@mail.neu.edu.cn
Subject: Signed Message
Content-type: text/plain; Charset = ISO-2022
Data: Sat.1 August 1998 13:07:48 + 0900
From 王丽娜 = <lnwang@mail.neu.edu.cn>

----BEGIN PRIVACY-ENHANCED MESSAGE----
Proc-Type:4;MIC-CLEAR
Content-Domain: FRC822
Originator-Certificate:
MIIBxzCCAWUCAg09MAOGCSqGSIb3DQEBAgUAMD4xCzAJBgNVBAYTAkpQMQOwCwYDVQQKEwRXSUF
AwHgYDVQQLExdDZXJOaWZpY2F0aW9uIFF1dGhvcmI0eTAeFw05NTA1MTYwNDMyMjJaFw05NzA1MTYw
yMjJaMIGRMQswCQYDVQQGEwYKUDEfMB0GA1UEChMWTkVDIEIuZm9ybWF0ZWMgU3IzdGVtczEjMCEG
ECxMaQWR2YW5jZWQgA1UEChMWTkVDIEIuZm9ybWF0ZWMgU3IzdGVtczEjMCEGA1UECxMaQWR2YW5
QgVGVjaG5vbG9neSBTeXN0ZW0xPDASBgNVBAMTC0IOQU1VUqEgWW91MCYGCSqGSIb3DQEJARYZaW5h
VyYUBhdHMubmIzLn5IYy5jby5qcDBcMA0GCSqGSIb3DQEBAQUAA0sAM1gCQQCjU9yiyHq3/ E-
BCYSXw0Y3itE
7ym +
dQgi2MLxCy550sr48F2wiqiLTemV0Tm2f4mE3hJIbI3YrkoI1IK9DngfUDAgMBAAEwDQYJKoZIhvcNAQBEC
BQADTQBZrksfauX9HEaS9rS4qn1goTJNHMFFgKxEF6Zx1N82VkXm9mqrh1iGJaf1Zoie2Z00QKqhYQCSc
+ IIqY1C
Key-Info: RSA,
fZCswfof3R0mQnhZQiKW
aMfu/3T2BsgCHXfJSASFCbuSEUBEvv8IHkcdbPw + yNiZ4Q01OGXtCGXIg45dryeIDXLCAUOkzZys7
Issuer - Certificate:
MIIBfjCCARwCATwwDQYJKoCIhvcNAQECBQAwPjELMAkJA1UEBhMCSIAxDTALBgNVBAoTBFdJREUxIDA
VBAsTF0NIcnRpZmIjYXRpb24gQXVOaG9yaXR5MB4XDTkzMTEwOTAwMDAwMFoXDTk2MTEwOTAwMDA
wPjELMAkGA1UEBhMCSIAxDTALBgNVBAoTBFdJREUxAIDAeBgNVBAsTFONIcnRpZmIjYXRpb24gQXY0aC
XR5MGgwDQYJKoZIhvcNAQEBBQADVwAwVAJNAG9zTm8/ Lpo3bvxog72830dT0oSPL2v7Vf9bqQQ
+ INVW5pZy
TQnyH8hEbTSCIn/
Vdjdph6j10k43WcPtSthSB4ggysVGXdHW3YQTAkCAwEEATANBgkqhkIG9w0BAQIFAANADu
EnfRO5r342YWhnYyn6ETxIU6dzd4JMbKgTzOrp0hQQtgF/ 9HxkhjE49NUP00VerZPKQUu7ppt2 + v/
ZFahDIOIhXQ
MIC-Info: RSA-MD5, RSA
13J8gUWLj
8ULYYZGqR7P2a8/ aaKs2PLU6/ mwj645CnVrdqiCos/ A3BXKB + 31mZYfV2nBELHD3Y + 0j3 +
1rhGGdoaGJUR1
e3v7tKStDDfvk1
Recipient-ID-Asymmetric:
MD4xCzAJBgNVBAYTAkpQMQ0wCwYDVQQKEwRXSURFMSAwHgYDVQQLExdDZXJ0aWZpY2FoaW9uIEF
x10eIQfs=
Key-Info: RSA,
H9dEUN53Em1bR3n3MpH1wGNto1h1T96LJIu5t92o ) QBzbop/
SAI017J64gwX5mXLBmYz09cEYUtowhwv9rnJ32
= =
```

图 6.32 加密后的信息

DEK-Info: **这个域表示信息本体的加密算法名称和十六进制初期信息。**

Recipient-ID-Asymmetric: **这个域表示受信者名称。**

Key-Info: **这个域表示传送给受信者的用公开钥加密了的** DEK **信息和使用的加密算法名称。**

其中,Recipient-ID-Asymmetric **和** Key-Info **两个域仅在受信者一方存在。**

2. 密钥管理方式

PEM 中的密钥管理采用如图 6 .33 所示的层次性的认证证书(certificate authority, CA)机制。

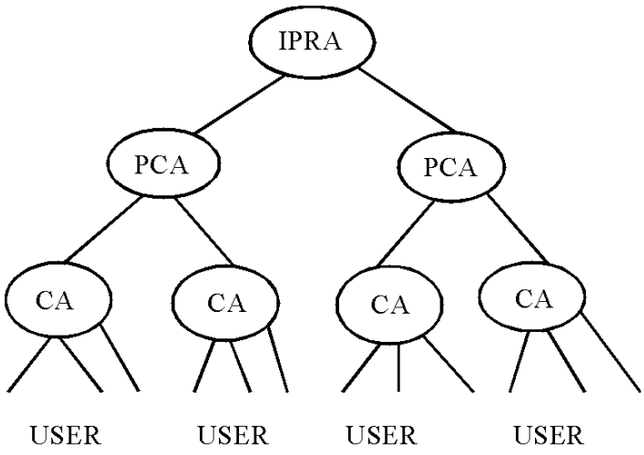


图 6 .33　层次性公开钥管理机制。IPRA 代表 Internet 方针注册局(IP register authority), PCA 代表方针认证证书,CA 代表认证证书, USER 代表用户。

为了使用 PEM,必须发行 CA 证明书,为了发行各个 PCA,CA 等证明书,必须采取一定的方针,这个方针公开对满足条件的用户和 CA 发行证明书。所谓证明书,就是保证用户和他的公开钥捆在一起的数据,实际上是用发行者的秘密钥将用户名、证明书和公开钥等的信息情报加密的内容。证明书内容如图 6 .34 所示。

对于证明书,用发行者的公开钥解密,就可以知道其中的用户名和公开钥。从非对称钥方式的性质可以知道: 能保证进行加密处理的人就是发行者本人,也能保证得到的用户名和公开钥就是所需要的用户名和公开钥。

证明书内容具体包括有效期限、发行者(对证明书加密的秘密钥的持有者)的信息、用户信息、用户公开钥信息。

用 IPRA 发行者的公开钥对 PCA 证明书密文(已经由 IPRA 发行者使用他的秘密钥对 PCA 证明书签名加密过)进行解密,获得 PCA 发行者的公开钥。用所获得的 PCA 发行者的公开钥对 CA 证明书进行解密,获得 CA 发行者的公开钥。用该公开钥对用户 A 证明书进行解密,获得用户 A 的公开钥。其过程如图 6 .35 所示。

最后给出在 Internet 上使用 E-mail 的几条衷告:

· 及时删除已经阅读过而且不需要保留的信息,当收到信息时,若认为有价值,可以将它们拷贝到自己的软盘上,否则难以避免他人阅读。

序列号 No = 4fb

有效期限 Validity: **从** 970721080522
到 980721080522

发行机关信息 issuer:
C = cn
O = WIDE
OU = Certification Authority

用户信息subject:
C = cn
O = Northeastern University
OU = Computer Cryptography Center
CN = Lina Wang
EmailAddress = lnwang@mail .neu .edu .cn

签名 Signature: **采用** md2 **哈希函数的** RSA **加密算法**

公开钥 publickey:
L = 512
N = b956dca2c87ab7fc40426125f0398de2b55hy29be750822d8c2f10afe79
d2caf8f05db08aa88b4de9953939b67h6564de1248c8dd8ae4a25d652fge78
1f342dskdsk334skkdfxa00clflkd0fmckow957dna3fbdcs78s922nsb26sajk
E = 10001

图 6 .34 **证明书**

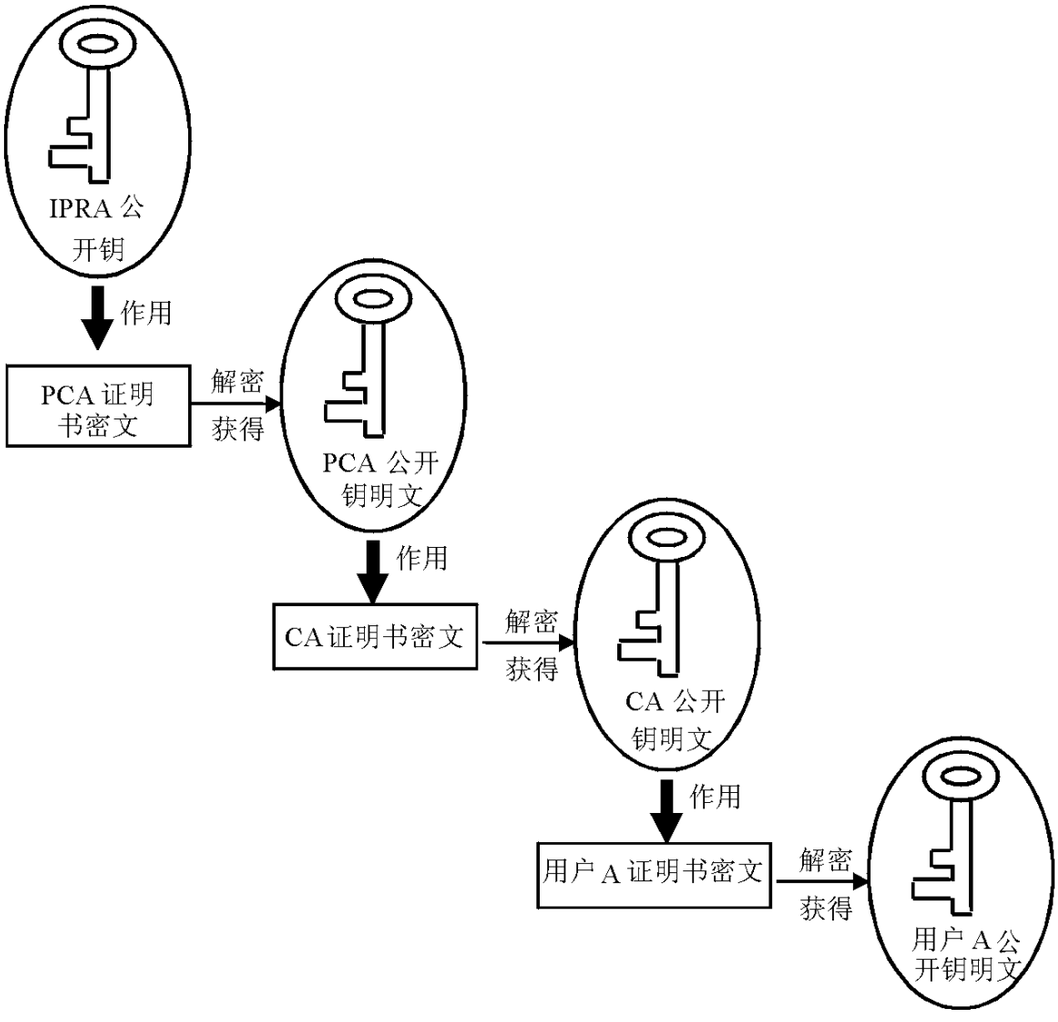


图 6 .35 **用户公开钥的解密方法**

· 将敏感的信息加密发送,使用 PEM,PGP(pretty good privacy)免费加密软件,确保信息安全。

· 回复信件时,一定要确认一下谁是始发者,以免回信发给清单上的一组人。

· 一旦发现邮件丢失、篡改等问题,及时与账号服务商联系,以求及时解决。

第 7 章　计算机病毒理论

随着计算机技术迅猛发展与普及,计算机已经成为当今信息社会必不可少的现代化工具,发挥着越来越重要的作用。系统分析与设计人员,在各种信息系统的建立与实施过程中,始终离不开计算机的应用。然而,人们在感受计算机强大功能的同时,也时时受到计算机病毒的威胁,它像一个幽灵暗中滋生并快速传播蔓延,使众多计算机数据文件受到侵袭,甚至造成系统的瘫痪。这不仅使广大计算机用户蒙受了巨大的经济损失,而且也给国家和社会带来了严重的危害。因此,学习和掌握计算机病毒理论与防治技术,成为了人们迫切的需要,受到了广泛的重视。本章首先概述计算机病毒的基本概念,然后介绍病毒的检测与分析技术,最后结合目前使用的典型杀毒软件,讲解防治和根除计算机病毒的实用方法。

7.1　计算机病毒的基本概念

计算机病毒(Virus)是隐藏在计算机系统中的程序,它不仅能够破坏计算机系统的正常运行,而且还具有很强的传染性。这种现象如同生物体感染了病毒一样,也同样具有自我繁殖、相互传染、激活再生的特征。计算机一旦接触到病毒,轻者影响计算机系统的性能,降低工作效率,重者可以毁坏计算机系统内部信息,并且迅速传播危害整个网络系统,造成停机。

1. 病毒的产生

计算机病毒大约出现于 20 世纪 60 年代末 70 年代初,关于它的起因有种种说法,下面介绍几种常见的病毒起源说。

1) 恶作剧起源说

有人认为,计算机病毒起源于恶作剧。通常,恶作剧者都认为自己有高超的技术和过人的智慧,他们非常熟悉计算机操作系统和编程技巧。为了显示自己在计算机方面的天资,设计开发出隐藏在计算机系统内部,能通过载体进行传播和复制,并且在一定条件下被激活的程序,这就是计算机病毒。例如:小球病毒,感染该毒后,计算机屏幕总是跳动的小球,无法正常工作。

2) 游戏程序起源说

持有这种观点的人认为,最早的计算机病毒,起源于程序员业余消遣所编的游戏程序。当时,贝尔实验室(Bell Laboratories)工作人员,休闲时间为了娱乐,在自己的计算机上编制可以吃掉对方程序的程序,这种具有很强的刺激性,被称为“核心战”游戏的程序,很接近今天我们所说的计算机病毒,人们把它称之为计算机病毒的雏形。

3) 科学幻想起源说

这种起源说认为,计算机病毒是从科幻小说中诞生的。1975 年,美国科普作家约翰 . 布伦纳(John Brunner)出版了一本名为《震荡波骑士》的幻想小说,该书以 Worm 和 Virus 为主人公,分别代表正义和邪恶的双方, 利用计算机进行斗争的故事。此后, 美国另一位科普作家托马斯 杰 .瑞思(Thomas J .Ryan)又出版一本名为《P-1 的青春》的科幻小说,在书中描写了一种特殊的,能够自我复制的计算机程序,该程序从一台计算机传播到另一台计算机,最后控制 7000 多台计算机的操作系统,造成一场灾难。作者将该程序称之为计算机病毒。因此,科幻小说也就成为了计算机病毒制造者的思想源泉。

4) 软件自我保护起源说

计算机软件是一种高科技的产品 ,其设计开发人员为此付出了巨大的脑力劳动,应该受到法律的保护。但是,总有一些违法分子,不劳而获,大量进行非法复制活动,从中牟取暴利。软件制造商为了保护自己的合法权力,在自己的软件产品中加入了破坏性的程序,以惩罚非法拷贝者。著名的巴基斯坦病毒(PakistainBrain)就是为此目的设计的,后经过多次修改,具有极强的破坏力。

关于计算机病毒起源的众多说法,都有一定的道理。但归根结底,来源于计算机系统本身所具有的动态修改自我复制能力,它成为了计算机病毒产生的瘟床。了解计算机病毒的历史,使我们能更有针对性的消灭它。

2. 病毒的特征

目前,已经发现的计算机病毒达几千种,它们虽然在产生的方式、破坏的程度上等各不相同,但是其本质特点却非常相似,概括的说,计算机病毒具有以下特点。

1) 破坏性

计算机病毒的主要目的是破坏计算机系统,使系统的资源和数据文件遭到干扰甚至被摧毁,根据其破坏系统程度的不同,可以分为良性病毒和恶性病毒。前者侵占计算机系统资源,使机器运行速度减慢,带来无谓的消耗;后者毁坏系统文件,造成死机,使系统无法启动。

2) 传染性

如同生物病毒一样,传染性是计算机病毒的重要特性。计算机病毒传播的速度很快,范围也极广,病毒一旦侵入主机,就立刻从一个程序传染到另一个程序,从一台机器传染到另外一台机器,再从一个网络传染到另外一个网络,可是,其分布是以几何级数增长的。

3) 隐藏性

计算机病毒虽然是一个程序,但它并不是一个独立存在的文件,病毒程序总是隐藏在其他合法文件或程序之中,而不容易被发现,使用户察觉不到,难以预料。这样才能达到非法进入系统,进行破坏的目的。用户一旦发现病毒,系统实际上已经被感染,资源及数据可能已经损坏。

4) 可激活性

计算机病毒的发作要有一定的条件,例如: 特定的日期,特定的标识符,使用特殊的文件等,只要满足了这些特定的条件,病毒就会立即被激活,开始破坏性的活动。

5) 针对性

病毒的编制者往往有特殊的破坏目的,因此不同的病毒,攻击的对象也不同。例如,有针对 APPLE 公司的 Macintosh 机器的,有针对 IBM 公司 PC 系列机及其兼容机的,有传染 command .com 文件的,也有传染扩展名为 .com 或 .exe 可执行文件的。

3. 病毒的分类

由于计算机病毒种类繁多、复杂,按照不同的方式,可以有许多的分类方法,下面介绍几种常用的分类方式及划分方法。

1) 按攻击的对象分类

按照计算机病毒攻击对象的不同,可以分为攻击微型机、攻击小型机、攻击大型机、攻击计算机网络四种。其中每种还可以进一步详细划分。例如,攻击微型机又分为攻击 IBM-PC 系列计算机和攻击 Macintosh 系列计算机等多种类型。目前,我国发现的病毒绝大部分是攻击微机 IBM-PC 系列及兼容机的。

2) 按寄生的方式分类

如同生物病毒一样,计算机病毒也有赖以生存的环境,依附于一定的载体。病毒攻击计算机系统后,其寄生的环境称之为宿主。按照计算机病毒在宿主环境中寄生的方式不同,可以分为覆盖式寄生病毒、代替式寄生病毒、链接式寄生病毒、添充式寄生病毒和转储式寄生病毒五种。

· 覆盖式寄生病毒是将自身程序代码部分或全部的覆盖在宿主程序上,从而使合法程序的部分功能或全部功能被破坏。

· 代替式寄生病毒是将自身程序代码替代宿主程序代码,从而使病毒程序以“合法”身份运行,完成其简单功能。

· 链接式寄生病毒是将自身程序代码附加在宿主程序代码之后,也可以是首部或中间,它并不破坏原合法程序。

· 添充式寄生病毒是将自身程序代码侵占宿主程序的空闲存储空间,它并不改变合法程序自身的长度。

· 转储式寄生病毒是将宿主程序代码改变存储位置,病毒程序自身代码侵占原合法程序的存储空间。

3) 按传染的方式分类

传染是计算机病毒的主要特征,按照其传染方式的不同可以分为传染磁盘引导区、传染可执行文件和综合型传染三种。

· 传染磁盘引导区,是指传染载体程序所在磁盘的引导区记录。若是用 DOS 系统对磁盘进行的格式化,对软盘而言,引导记录是 DOS 的 BOOT 区引导程序;对硬盘而言,引导记录有硬盘主引导程序和硬盘 DOS 分区中 BOOT 区引导程序。传染磁盘引导区病毒,将系统原有的引导记录隐藏在磁盘的其他空间,从而使其在开始运行就获得了系统的控制权。

· 传染可执行文件是指病毒以链接方式,寄生在系统隐含文件上。当该文件被调用执行时,病毒程序就获得了控制权,开始其破坏活动。

· 综合型传染是指既传染磁盘引导区程序,又传染系统文件的综合病毒。例如,HIP病毒不仅传染 command .com 及可执行文件,而且还传染磁盘主引导区,这种病毒用 FORMAT 命令格式化硬盘都不能消除,给杀毒带来困难。

4) 按侵入途径分类

根据计算机病毒侵入系统的途径不同,可以分为源码病毒、操作系统病毒、入侵病毒和外壳病毒四种。

· 源码病毒是指病毒在源程序被编译前就被插入到源程序中,然后被编译成合法程序的一部分。源码病毒通常攻击高级语言编写的程序。由于制造这类病毒难度较大,所以感染的范围也有限。

· 操作系统病毒是指病毒程序将自身加入或替代操作系统工作,这种病毒最常见,危害也最大。例如,小球病毒就属于典型的操作系统病毒。

· 入侵病毒是将自身程序侵入到主程序之中,清除这种病毒的同时,也就破坏了系统中的主程序。

· 外壳病毒是将自身程序放在主程序的周围,一般不对原来的程序进行修改。这种病毒容易编制,大约占病毒总数的一半。

7.2 计算机病毒的分析

初步了解计算机病毒的基本概念之后,预防和消除病毒,成为广大计算机用户的首要任务。为此,对计算机病毒进行分析,掌握病毒的破坏现象、表现症状、程序结构及检测方法。

1. 病毒的破坏现象

计算机病毒的出现,从不同程度上反映出计算机系统的脆弱性,从而使病毒对系统攻击破坏成为可能。通常计算机病毒的破坏现象表现在以下几方面。

1) 破坏文件分配表,造成磁盘信息的丢失。对于这种破坏,用 DIR 命令有时发现不了,当用户使用 DIR 命令查看磁盘目录时,文件名还存在,但是文件的本体与文件名失去了联系。

2) 删除磁盘上可执行的文件或数据文件。如果被删除的是系统引导文件,则导致该磁盘无法启动系统。

3) 修改或删除文件中数据信息。随着信息技术的广泛普及,电子银行、电子商务应用越来越多,这种破坏给信息系统造成致命打击。

4) 对整个磁盘或磁盘的部分磁道、扇区进行格式化。

5) 在系统中产生新的文件,而这些文件对用户而言,有时是不可见的。

6) 改变文件的属性、建立日期,增加文件的长度,使系统运行速度减慢。

7) 更改磁盘卷标,改变磁盘上程序信息的存储状态。

8) 侵占磁盘的存储空间,改变磁盘的分配,或增加磁盘的坏扇区,造成数据的丢失或写入错误。

9) 影响内存中常驻程序的正常运行。

10) 破坏屏幕的正常显示,出现异常图形、符号、文字信息。

11) 系统空挂,可以造成屏幕或键盘的封锁状态。

12) 硬件接口异常,软盘磁头来回移动。

13) 影响系统正常启动,破坏系统正常运行。键盘锁定。

2. 病毒程序结构

计算机病毒程序是为了特殊目的而编制的,它通过修改其他程序而把自己复制进去,并且传染该程序。一般来说,计算机病毒程序包括三个功能模块: 引导模块、传染模块和破坏模块,这些模块功能独立,同时又相互关联,构成病毒程序的整体。

1) 引导模块

引导模块的功能是借助宿主程序,将病毒程序从外存引进内存,以便使传染模块和破坏模块进入活动状态。另外,引导模块还可以将分别存放的病毒程序链接在一起,重新进行装配,形成新的病毒程序,破坏计算机系统。

2) 传染模块

传染模块的功能将病毒迅速传染,尽可能扩大染毒范围。病毒的传染模块由两部分组成: 条件判断部分和程序主体部分,前者负责判断传染条件是否成立,后者负责将病毒程序与宿主程序链接,完成传染病毒的工作。

3) 破坏模块

病毒编制者的意图,就是攻击破坏计算机系统,所以破坏模块是病毒程序的核心部分。破坏模块在进行各种攻击之前,首先判断破坏条件是否成立,只有条件全部满足时,破坏模块才开始其破坏活动。

总之,病毒程序各功能模块,相互依赖,协调进行,引导模块是传染和破坏模块的基础,破坏模块又依靠传染模块,扩大攻击范围,完成对计算机系统及其数据文件的破坏工作。图 7.1 给出病毒程序工作的流程图。

3. 病毒的症状

计算机病毒通常在发作前会尽可能广为扩散。感染病毒后的系统,会表现出一些异常症状。病毒感染的症状一般会在它造成危害之前表现出来,如果能准确地识别它,就可以抓住时机,在病毒造成破坏之前发现并消除它。下面给出感染病毒的具体症状。

1) 计算机屏幕显示异常

例 7.1 屏幕出现异常滚动。当小球病毒发作时,屏幕上就会出现一个小球,不停的无规则的运动,碰到屏幕边缘或英文字母就反弹回来,在汉字系统下,会破坏汉字的显示,使屏幕内容变得面目皆非。

例 7.2 屏幕出现异常提示信息。大麻病毒在系统启动时提示:“ Your PC is now stoned !”,LEGALIZE MARIJVANA (现在你的 PC 机被击中了,合法大麻)。

磁盘杀手(DISK KILLER)病毒在破坏条件成立时显示:

Disk killer - Version 1 .00 by computer 04/ 01/ 1989

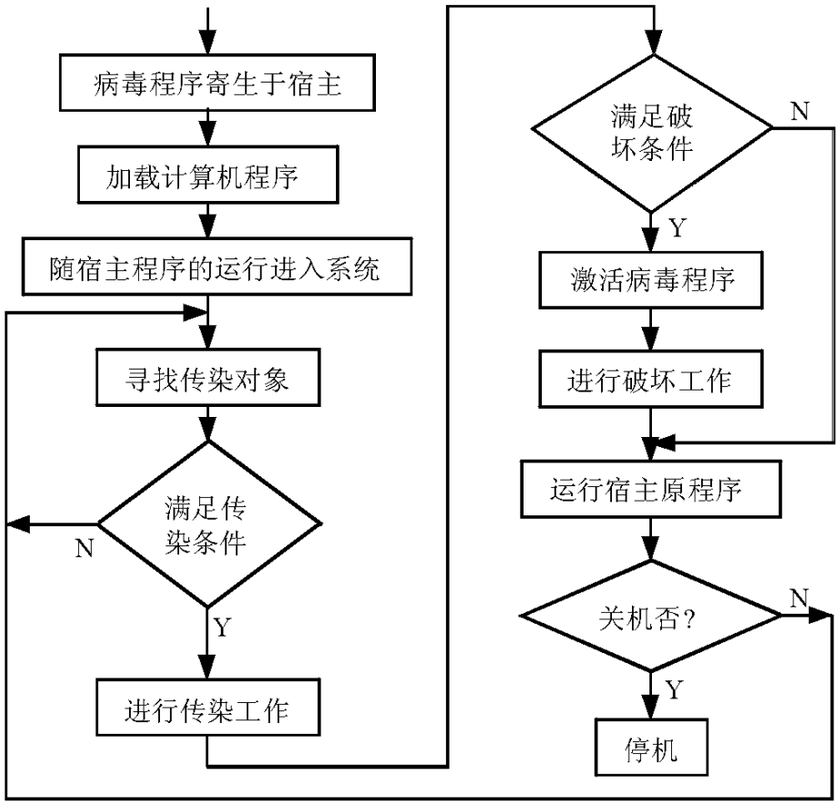


图 7.1　计算机病毒程序流程图

PROCESSING

WARNING !!

Don't turn off the power or remove the diskette while disk killeries processing PROLESSING

Now you can turn off the power

I wish you luck

与磁盘杀手病毒类似,许多病毒在发作时都有提示信息。

例 7.3　屏幕出现异常图形显示。黑色星期五病毒发作时,屏幕上的部分内容会发生滚动,出现一个长方形亮条。1575 病毒发作时,屏幕出现小毛虫。

2) 计算机系统的蜂鸣器出现异常声响

例 7.4　YANKEE DOODLE **病毒进入内存后,下午** 5:00(17:00 **点)准时奏出扬基歌主旋律。**

3) 计算机系统的运行速度异常减慢。通常,所有的计算机病毒发作时,都会造成系统运行速度下降,这是病毒存在较为明显的症兆。

4) 计算机系统出现异常死机。例如 Brain, Lehigh **等病毒都会造成系统突然死机,另外值得注意的是一些病毒在传染失败时,也将导致系统异常停机。**

5) 系统文件的长度发生改变。通常,破坏可执行文件的病毒都会修改文件的字节长度。例如:黑色星期五病毒使 .COM **文件中的长度增加** 1813 **字节;** YANKEE DOODLE **病毒使可执行文件长度增加** 2885 **字节。**

6) 数据文件的内容被修改或删除。一些病毒在发作时改名或删除染病文件,造成文件丢失。

7) 计算机系统的存储容易异常减少。因为病毒本身有复制能力,传染一个宿主程序

后,还会反复传染,造成储存系统的存储容量迅速减少。另外,多个病毒同时传染一个宿主后也能造成磁盘容量的迅速减少。

8) 在键盘上敲入的字符和屏幕上显示的字符不一致。键盘锁定或修改功能键,造成混乱。

9) 硬件的接口异常。例如:UNPRINTING 病毒传染系统后,只要病毒的计数器为 224 时,系统就不能通信了,并且打印机也停止工作,出现"No Paper"提示信息。

10) 中断向量发生变化。许多病毒通过中断获得系统的控制权,从而破坏系统和数据文件。

计算机病毒多种多样,而且,新的病毒总在不断的出现,我们很难通过病毒的个别症兆,而判定病毒的种类。但是,掌握了计算机系统这些异常现象后,当机器出现这些症状时,千万警惕病毒的存在,尽快寻找最佳措施消除病毒。

4. 病毒的检测

由于计算机病毒的危害极大,迫使各国计算机专业人员投入大量的人力物力进行病毒的研究工作。为了有效防治计算机病毒,首先应该掌握病毒的识别和检测技术,尽早发现病毒,及时清除掉,就可以避免病毒发作时造成的严重损失。目前,研究人员根据计算机病毒程序的结构,对其潜伏机制、再生机制和激活机制进行了深入的分析,在计算机病毒检测方面取得了很大进展。可以预计凭借先进的软件手段和丰富的观察经验,一定可以检测到系统中存在的未知病毒。下面分几个方面概括一下计算机病毒的检测方法和过程。

1) 通过对计算机病毒标识符的查找,可以检测病毒的存在。计算机病毒对磁盘或文件进行传染后,一般都要在该磁盘或宿主程序上给出自己的病毒标识符,其作用是使病毒自身能认识自己。病毒标识符都是由 26 个英文字母和数字组成的字符串,它位于程序的某个特定位置。例如:黑色星期五病毒以 Msdos 为其标识符;Liberty 病毒传染 .COM 文件时,标识符为 Liberty,传染 .EXE 文件时,标识符为 FFFFH,因此,查找到这些病毒的标识符,就可以检测到是何种计算机病毒存在。

2) 通过观察到的计算机系统各种异常现象,检测病毒的存在。一般病毒程序都有一定的潜伏期,即计算机病毒传染宿主程序之后,到其表现部分与破坏部分被激活而没有被用户发现的一段时间。计算机病毒在潜伏期中,使计算机系统在正常工作中出现种种异常现象(前边已经介绍的病毒症状),出现这些异常现象后,用以下实用方法可以检测到病毒的存在。

(1) 检测系统的内存容量。利用系统命令 CHKDSK 调试程序 DEBUG 和工具程序 PCTOOLS能够得到系统的内存容量报告。若内存容量和系统无病毒常驻时的结果相比减少了,说明可能有病毒存在。用系统提供的 CHKDSK .COM 程序检测内存具体方法如下。

```
C > CHKDSK
    33462272    bytes    total    disk    space
       53248    bytes    in    2    hidden    files
```

```
      4096    bytes    in   1   directiries
   5171200    bytes    in   75   user   files
  28233728    bytes    available on disk
    655360    bytes    total memory ;全部内存容量
    595296    bytes    free ; 剩余内存容量
```

上面显示系统无病毒常驻内存容量,如果有 Brain 病毒常驻内存,用 CHKDSK .com 程序检测结果为:

```
C > CHKDSK
  33462272    bytes    total   disk   space
     53248    bytes    in   2   hidden   files
      4096    bytes    in   1   directiries
   5171200    bytes    in   75   user   files
  28233728    bytes    available on disk
    648192    bytes    total memory
    588128    bytes    free
```

由以上显示报告可以发现,系统内存容量减少了 7K 字节。于是就可以检测到病毒的存在。但是有些病毒常驻内存后,并不减少内存的容量,这样单纯使用内存容量检测,还不能准确判断是否有病毒,应该结合其他检测方法。

(2) 检测中断向量表。计算机病毒传染系统和驻留内存一般都与计算机系统中断服务程序有关,因此,了解计算机系统的有关中断对于检测病毒的存在十分有益。为了提高 CPU 工作效率,主机与外部设备都是通过中断方式进行联系的。以 DOS 系统为例,系统中断属于向量中断,系统可处理 256 种中断,中断号的范围为 0～255,中断处理程序的入口地址组成中断向量表,中断向量表位于内存中从 0000H～03FFH 的空间内。从目前发现的病毒来看,大多数病毒都是通过一些中断来获得系统控制权的。于是,修改中断向量,对中断向量表进行操作,是病毒程序必作的工作。表 7.1 给出病毒程序经常篡改的中断向量表。

表 7.1　部分中断向量地址表

性　质	中断号	向量地址(十六进制)	功能说明
外部中断	08	0000∶0020-0000∶0023	计时器中断
显示中断	10	0000∶0040-0000∶0043	视频 I/O 调用
服务中断	13	0000∶004C-0000∶004F	磁盘 I/O 调用
键盘服务软中断	16	0000∶0058-0000∶005B	键盘 I/O 调用
时钟中断	1A	0000∶0068-0000∶006B	日期/时间调用
特殊中断	1C	0000∶0070-0000∶0073	计时器中断控制
系统中断	21	0000∶0084-0000∶0087	通用功能调用
地址中断	24	0000∶0090-0000∶0093	标准错误向量
系统中断	25	0000∶0094-0000∶0097	绝对磁盘读
系统中断	26	0000∶0098-0000∶0098	绝对磁盘写
退出驻留中断	27	0000∶009C-0000∶009F	程序结束且保留内存

所以,将中断向量表的内容进行保存,经常将新老向量表加以比较,就容易发现病毒的痕迹,从而检测到病毒的存在。

(3) 检查软盘的引导扇区和硬盘的主引导扇区。软盘的引导扇区和硬盘的主引导扇区是系统病毒传染的对象,如果发现其中的内容与正常的引导记录不同,就可断定是病毒存在。软盘引导扇区的检查,可以通过 DEBUG 和 PCTOOLS 程序把扇区中的内容显示出来,进行分析判断。硬盘主引导扇区可以用 Norton 等工具查看,主引导扇区中一般含有"Partition Table"字符串,而病毒传染后的主引导区不含有该字符串,于是说明感染了病毒。

(4) 检测文件的属性及长度。计算机病毒在传染文件时,往往更改计算机文件的建立时间、日期、数据长度,这些内容可以通过系统内部命令 DIR 显示,同时也可以通过 PCTOOLS工具检测到。表 7.2 给出病毒传染文件后,文件增加的长度。

表 7.2 病毒传染文件后,文件增加的字节数

病毒名称	感染病毒后文件增加字节数
AIDS,艾滋病	13312
Alabama,变形虫	1392
Anthrax,炭疽病	1206
April Frist,愚人节	897
Austrian,奥地利	648
Best wishes,最美好的祝愿	1024
Black Monday,黑色星期一	1055
Bouncing ball,跳动的球	1024
China bomb,中国炸弹	1492
Christmas boot,圣诞节 boot	512
Diana,戴安娜	1805
DBASE,数据库	1864
Disk killer,磁盘杀手	3072
Friday 13^{th},星期五 13 日	1808
Hebrew University,希伯来大学	1808
Icelandic3,冰岛 3	848
Jerusalem USA,耶路撒冷美国	1813
Leapfrog,跳蛙	516
Music bug,音乐臭虫	4608
Marijuana,大麻	512

如果通过比较,发现文件莫名其妙加长了,可以断定有计算机病毒存在。

3) 通过实用的病毒检测软件查毒。目前,有许多十分流行的病毒检测软件,经实践证明可以检测计算机的绝大多数病毒。例如:美国 MCAFEE ASSOCIATES 公司开发研制的 SCAN .EXE 就是一个可靠的实用软件。当运行病毒扫描程序 SCAN .EXE 时,它开始扫描计算机系统和磁盘,并且能识别所有已经存在于 PC 机上的传染病毒,它能够指出被传染的文件和系统区域,并识别出病毒的种类。一般检测单个磁盘用 SCAN D: 就可以了,其中 D 代表任意盘符。程序执行时,在屏幕上随时显示正在检测的部位,其检测顺序是先 RAM 区,再 BOOT 区,接着对盘上逐个文件进行检测,最后将检测结果给用户,下面给出一个检测实例:

```
A > SCAN B:
scan 1 .8v52 copyright 1989 by Mcafee Associates
scanning B: LCH .com
Found Yankee Doodle Virus
Scanning B: c6 .exe
Found Yankee Doodle Virus
Disk B: contain 1 directories and 7 files
2 files contain viruses
```

说明有两个文件染上 Yankee Doodle 病毒。除此以外,Central Point 公司推出 CPAV 软件,Engineering 公司推出的 TNTVIRUS .exe 以及一般监视程序 Generic monitoring programs 和工具 NORTON7 .0 程序都是很好的计算机病毒检测和消除软件。上网的用户,还可以通过 Internet 网下载消毒剂 Disinfectant 程序,这个免费软件能帮助你清洁计算机系统。它扫描你的资源,当发现可疑的病毒时就通知你,起到资源保护作用。

7.3 计算机病毒的防治

在对计算机病毒进行了理论与技术分析之后,使我们对病毒的攻击对象、破坏现象、程序机理有了进一步的了解,更加清醒的认识到计算机病毒对信息系统的危害性。因此,必须采取有效措施,防治计算机病毒的感染与发作。这部分将对如何防范计算机病毒、清除病毒的技术手段和目前流行的杀病毒软件及使用方法进行介绍。

1. 病毒的防范

计算机病毒的防御措施,应该包括两重含义,一是建立法律制度、提高教育素质,从管理方法上防范;二是加大技术投入与研究力度,开发和研制出更新的防治病毒的硬件、软件产品,从技术方法上防范。只有将这两种方法结合起来考虑,才能行之有效地防止计算机病毒的传播。

1) 管理方面的预防

· 尊重知识产权,不要随意拷贝和使用未经安全检测的软件,杜绝计算机病毒交叉感染和传播渠道。

· 对于新购置的计算机系统硬、软设备,都应该首先进行病毒检查,最好保证不外借。

· 慎重使用网络和公告牌信息,注意其规范性。

· 尽量禁止在计算机上运行任何游戏。游戏盘使用频繁,最容易感染病毒。

· 对于系统之中的重要数据,最好不要存储在系统盘上,并且随时进行备份。

· 最好不要用软盘引导系统,这样可以较好的防止引导区传染的计算机病毒的传播。

· 建立计算机系统使用登记表,详细记录机器管理者、使用者的情况,对重要的操作过程进行监督。

· 采取必要的病毒检测、监察措施,制定完善的管理准则。

· 加强教育和宣传工作,使广大的计算机专业人员都认识到编制计算机病毒软件是不道德的犯罪行为。从伦理道理和社会舆论上扼杀病毒的产生。

· 建立、健全各种法律制度,保障计算机系统的安全性。呼吁国际社会,制定严格的法律条文,对制造病毒者依法制裁,从根本上杜绝计算机病毒的来源。

总之,完善的管理制度,可以人为的根除或减少病毒的制造源。这在一定程度上反映出一种社会文明与文化发展的水平,对整个计算机技术的发展起到促进作用。

2) 技术方面的防御

计算机病毒的技术防御,包括软件预防与硬件预防两种,它是以计算机技术为基础的防御措施。所谓硬件防御是指通过计算机硬件的方法预防计算机病毒侵入系统,主要采用防病毒卡。防病毒卡作为 ROM 插件,在系统启动时就可获得控制权,使机器具有了免疫力,只要系统中运行的程序带有病毒,防毒卡就会发现,给出警告信息,并在内存中将其清除掉。所谓软件防御,是指通过计算机软件的方法预防计算机病毒侵入系统。这是较为常用和普及率较高的方法,目前,预防病毒软件很多,归纳起来,一般这类软件应包括下述几大功能。

· 监视常驻内存的程序。由于计算机病毒都是以常驻内存的形式传染系统,因此只要有效地监视任意常驻内存的程序,就可以防止大多数计算机病毒。以程序 MEMORY .C 为例,用户首先定义允许常驻内存的程序名单,当运行中发现有程序要常驻内存时,MEMORY .C就向用户提出警告:“Application trying to get TSR .permission granted (Y/N)”,若用户回答“Y”,则正常运行;否则,MEMORY .C 程序将重新启动。MEMORY .C 程序如下:

```
#pragma inline
#include <dos.h>
#include <bios.h>
#define u unsigned
void interrupt (*old21)();
void interrupt (*old27)();
void interrupt new21(u BP,u SI, u DS,u EX, u DX, u CX,u BX, u AX,u IP,u CS,u FLAGS);
void interrupt new27(u BP,u SI, u DS,u EX, u DX, u CX,u BX, u AX,u IP,u CS,u FLAGS);
char executing_file[50],ch,far *fname;
int i=0,first_time_flag=0;
char far *scr=(char far *)0xb8000000;
main()
{
  strcpy(executing_file,"Unknowed Applications.");
  old21 = getvect(0x21);
```

```
    old27 = getvect(0x27);
    setvect(0x21,new21);
    setvect(0x27,new27);
    keep(0,850);
}
void interrupt new21(u BP,u SI, u DS,u EX, u DX, u CX,u BX, u AX,u IP,u CS,u FLAGS);
{
if(_AH==0X31)
{
   if(first_time_flag)
   {
                tsrmessage();
                if(ch=='N'||ch=='n')
                CS=0xffff;
                IP=0X0000;
                return;
                }
   }
  else first_time_flag++;
}
if(_AH==0x4b)
{
   fname=MK_FP(DS,DX);
   i=0;
   while(*fname)
   { ch=*fname;
     executing file[i]=ch;
     fname++;
     i++;
   }
executing_file[i]='';
}
      asm pop bp
      asm pop di
      asm pop si
      asm pop ds
      asm pop es
      asm pop dx
      asm pop cx
      asm pop bx
      asm pop ax
      asm jmp cs:_old21
}
void interrupt new27(u BP,u SI, u DS,u EX, u DX,u CX,u BX, u AX,u IP,u CS,u FLAGS)
      {tsrmessage();
      if(ch=='N'||ch=='n')
      {      CS=0xffff;
```

```
            IP = 0x0000;
            return;
}
      asm pop bp
      asm pop di
      asm pop si
      asm pop ds
      asm pop es
      asm pop dx
      asm pop cx
      asm pop bx
      asm pop ax
      asm jmp cs:_old27
}
tsrmessage()
  {     fwrites(4,8,1,executing_file);
        fwrites(4,9,1,"Applications trying to get TSR .
                        Permission granted(y/n) . .");
      while(((ch=bioskey(0))!='y')&&(ch!='N')&&(ch!
                    ='Y')&&(ch!='n'));
      if(ch=='n'||ch=='N')
      {fwrites(4,12,1,"Now press any key t Reboot ....");
      bioskey(0);
      }
}
fwrites(curx,cury,attrib,st1)
int curx,cury,attrib;
char *st1;
{   int i;
    if(curx>=81||curx<=0)
       {  curx=0;
          exit(0);
    }
    if(cury>=26||cury<=0)
      {     curx=0;
            exit(0);
    }
    curx--;
    cury--;
    if(attrib==1) attrib=112;
    if(attrib==2) attrib=7;
    for(i=curx*2;*st1++,i++)
      {   *(scr+160*cury+i)=*st1;
        i++;
         *(scr+160*cury  +i)=attrib;
    }
}
```

·防止可执行文件被改写。因为传染可执行文件的计算机病毒,都要对该文件进行改写,所以若能防止病毒程序改写 .EXE 和 .COM 文件,就可能防止大多数文件感染病毒。以 C 语言和汇编语言混合编写的程序 NOOPEND .C,在系统运行时可以防止可执行文件被非法打开。当运行中试图打开 .EXE, .COM 和 .SYS 文件时,NOOPEND .C 程序向用户提示:“Application is trying open (EXE/ COM/ SYS) file inwrite / R + W mode”,“Permission granted ? (y/ n)”,或“Application is trying to rename(EXE/ COM/ SYS) file”,“Permission Granted ? (y/ n)”,用户根据系统具体环境,决定是否允许程序提出的情况。NOOPEND .C 程序如下:

```
#pragma inline
#include <dos.h>
#include <bios.h>
#define u unsigned
void interrupt (*old21)();
void interrupt new21(u BP,u SI, u DS,u EX, u DX,u CX,u BX, u AX,u IP,u CS,u FLAGS);
char executing_file[50],far *fname,ch,filename[50],apurva[20];
int i=0,violation;
char far *scr=(char far *)0xb8000000;
main()
{
   old21 =getvect(0x21);
   setvect(0x21,new21);
   keep(0,700);
}
void interrupt new21(u BP,u SI, u DS,u EX, u DX, u CX,u BX, u AX,u IP,u CS,u FLAGS)
{
  if(_AH==0x3d||_AH==0x3c)
  {
   fname=MK_FP(DS,DX);
   i=0;
   while(*fname)
   {ch=*fname;
   filename[i]=ch;
   fname++;
   i++;
   }
   fname[i]='';
   strupr(filename);
   i=0;while(filenmae[i]!='.')i++;
   if(!strcmp(&filename[i],".EXE")||
!strcmp(&filename[i],".COM")||!strcmp(&filename[i],".SYS"))
        {  if((AX & 7)||((AX/256)==0x3c))
           {   violation=1;
               tsrmessage();
               if(ch=='n'||ch=='N')
               {AX=5;
```

```
                    asm stc
                    asm pushf
                    asm pop FLAGS
                    return;
                    }
                }
            }
        }
if(_AH == 0x56)
    { fname=MK_FP(DS,DX);
      i=0;
      while(*fname)
      {   ch= *fname;
        filename[i]=ch;
          fname++;
          i++;
      }
    filename[i]='';
    strupr(filename);
    i=0;
    while(filename[i]!='.')i++;
    if(!strcmp(&filename[i],".EXE")||!strcmp(&filename[i],".COM")
    ||!strcmp(&filename[i],".SYS"))
            {       violation=2;
                    tsrmessage();
                    if(ch=='n'||ch=='N')
                {   _AX=5;
                    asm stc
                    asm pushf
                    asm pop FLAGS
                    return;
                }
            }
            }
        if(_AH==0x4b)
            {
            fname=MK_FP(DS,DX);
            i=0;
            while(*fname)
             {   ch= *fname;
            executing_file[i]=ch;
                fname++;
                i++;
          }
            executing_file[i]=ch;
        }
    asm pop bp
```

```
asm pop di
asm pop si
asm pop ds
asm pop es
asm pop dx
asm pop cx
asm pop bx
asm pop ax
      asm jmp cs:_old21
}
      tsrmessage()
      {       fwrites(4,8,1,executing_file);
              if(violation ==1)
                  fwrites(4,9,1,"Application is trying to open
              (EXE/COM/SYS) file in WRITE/R+W mode .");
              else fwrites(4,9,1,"Application is trying to
                      rename (EXE/COM/SYS)file .");
           fwrites(4,10,1,"Filename -");
           fwrites(13,10,1,filename);
           fwrites(4,11,1,"Permission granted(y/n) ...");
        while(((ch=bioskey(0))!='y')&&(ch!='N')&&(ch!
                          ='Y')&&(ch!='n'));
        if(ch=='n'||ch=='N')
        {fwrites(4,12,1,"Please do not run this application
                          till it is cleared of the virus .");
        fwrites(4,13,1,"Now press any key to continue ....");
        bioskey(0);
        }
}
fwrites(curx,cury,attrib,st1)
int curx,cury,attrib;
char *st1;
   {   int i;
       if(curx>=81||curx<=0)
          {          curx=0;
                     exit(0);
          }
       if(cury>=26||cury<=0)
          {       cury=0;
                  exit(0);
       }
       curx--;
       cury--;
       if(attrib==1) attrib=112;
       if(attrib==2) attrib=7;
       for(i=curx*2;*st1++,i++)
          {    *(scr+160*cury+i)=*st1;
```

```
i+ + ;
* (scr + 160 * cury + i) = attrib;
}
```

· 禁止程序直接写入磁盘引导区。因为对于传染引导区的计算机病毒,都要通过磁盘的绝对读写来完成,所以若能有效地防止程序直接写入磁盘,就可以阻止大多数计算机病毒的传播。另外,禁止磁盘的写操作、监视系统中断向量的修改,都能够在技术上防止病毒的侵入,保证计算机系统安全。

2. 清除计算机病毒的原则

无论多么严密的病毒防范措施,都无法绝对禁止计算机病毒的侵入,因此,一旦确定计算机系统感染了病毒,应该立即设法清除病毒,恢复系统,使计算机在安全环境下继续运行。当然,根据入侵病毒种类的不同,清除病毒的方法也不同,目前,国内流行的硬、软清除病毒工具很多,虽然它们在具体操作过程中采用不同的方法,但是它们却都遵循一定的原则。

1) 计算机病毒的清除工作最好在无毒的环境中进行,以确保清除病毒的有效性。这要求清毒前,用无毒的计算机系统引导盘重新启动系统;或者清除内存的计算机病毒,恢复正常的中断向量。

2) 把启动系统的系统盘和杀毒软件盘加上写保护标签,以防止其在消除病毒过程中感染上病毒。

3) 在清除病毒之前,一定要确认系统或文件确实存在病毒,并且准确判断出病毒的种类,以保证清毒的有效。否则,可能会破坏原有的系统和文件。

4) 尽可能地找出计算机病毒的宿主程序,确定其病毒标识符和传染对象,即搞清楚病毒传染的是引导区还是文件,或者是既传染引导区,又传染文件,以便找准清除病毒的最佳方法。

5) 不要使用激活病毒的方法检测病毒,因为在激活病毒的同时,计算机系统有可能已经被破坏了。

6) 清毒工作要深入而全面,为保证清除工作过程的正确性,要对检测到的病毒进行认真分析研究,尤其是对自身加密的病毒引起重视,把修改过的文件转换过来,否则清除病毒后文件不能运行。

7) 不能用病毒标识免疫方法清除病毒。标识免疫的方法是利用病毒进入系统的条件性来实现的,通常病毒在进入系统之前都要判断内存是否已驻留了病毒,如果是,则退出其加载过程。根据这一免疫原理,编写病毒标识存入内存,以防止病毒入侵。此方法理论上可行,但实际应用中却不保险,因为病毒变种繁多,流传的广而快,将引导区存入所有这些病毒的标识,也就无法起到引导系统的功能了。因此,免疫方法常带有很大的欺骗性,原则上不能作为清毒工具。

8) 对于那些既传染文件,又传染引导区的计算机病毒,在清除文件病毒之后,还应该清除引导区中的病毒代码。以防止这些病毒代码再次重新生成计算机病毒。

9) 在对文件的病毒清除之后,必须检查系统中其他同类文件中是否传染了此病毒,

以避免清毒后系统再次运行时,又出现此种病毒。

上述原则是计算机病毒清除工作中必须遵守的,根据这些原则,图 7.2 给出了清除计算机病毒的全过程。

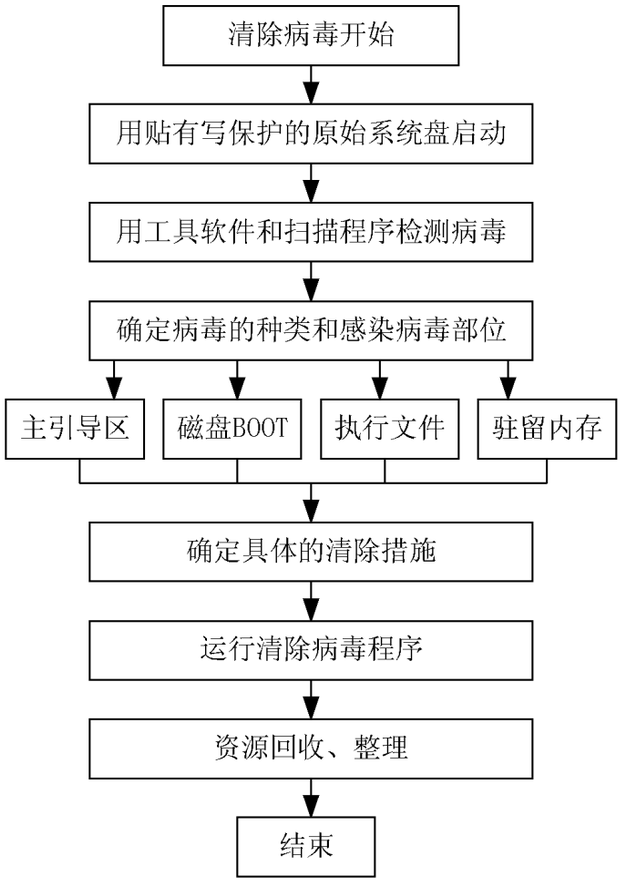


图 7.2　清除计算机病毒流程图

3. 常用杀毒软件介绍

计算机病毒的手工治理,难以适应面宽量大的病毒防治工作。而且手工方式对计算机用户的技术水平要求较高,稍有不慎就可能造成比病毒本身更大的损失,所以,手工消毒受到一定限制。为此,计算机病毒防治专家,开发研制出许多高效的消毒软件产品,用于 DOS 平台的病毒、Windows 平台的病毒和网络病毒的防治工作,这些杀病毒软件的应用,为计算机系统的安全运行起到了良好的保护作用。

1) 杀病毒软件 KILL

KILL 杀病毒软件是国家安全部开发的,它可以检测到 DOS 环境下的病毒,并且可以成功地清除掉这些病毒。该软件中的工作机理是对各种已知病毒结构、染病毒过程进行详细分析,据此建立起计算机病毒数据库,因此它能够准确地指出所感染病毒的名称和位置,然后对已经感染病毒的计算机系统和程序文件进行检测和可靠的恢复。KILL 对国产病毒特别有效,它操作简单,用户不需要对病毒有任何了解,一切判断处理工作均由软件自动完成,并且在计算机屏幕上显示详细的信息,因此适用于广大普通的计算机用户。

KILL 软件的使用方法:在 DOS 系统启动后,将 KILL 杀病毒盘插入 A 或 B 软盘驱动器,然后输入命令:

KILL[驱动器号:][路径名] [文件名]

其中[]中内容为可选项,表示对指定驱动器,指定路径下的指定文件进行杀病毒处理。KILL 主菜单如图 7.3 所示。

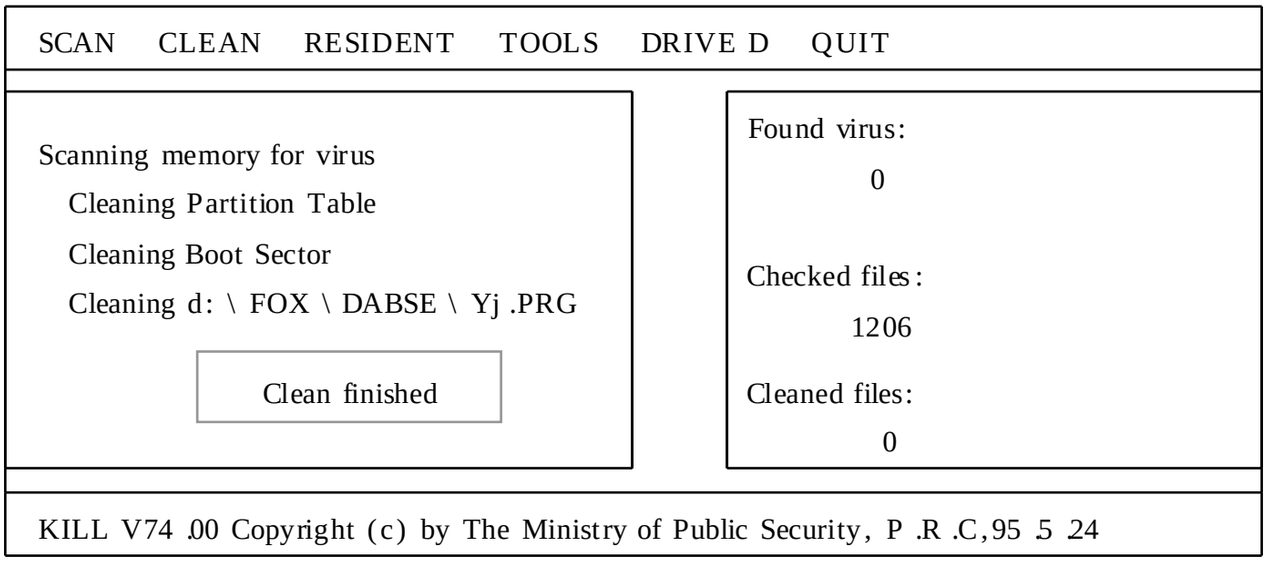


图 7.3 KILL 主菜单屏幕显示

图 7.3 中各功能项含义如下:

· SCAN 项: 只检查病毒是否存在,如果发现病毒,屏幕将显示被感染的文件名及病毒的种类:FIND VIRUS IN MEMORY, REBOOT SYSTEM !!!,但不进行杀病毒处理。

· CLEAN 项: 它将对病毒进行清除,屏幕将显示被清除的病毒种类和被感染的文件名,例如:Found DIR-2 Virus, Killed ! 病毒清除完毕后,则显示:Cleaning finished。

· RESIDENT 项: KILL 软件常驻内存,用户暂不使用该功能。

· TOOLS 项: 软件辅助分析工具,用户暂不使用该功能。

· DRIVE 项: 该功能可以改变原来指定的驱动器名、路径名和文件名,而不必退出 KILL 重新运行。选中此项后,屏幕会出现一个对话窗口,用户可输入新选的驱动器名、路径名和文件名。

· QUIT 项: 选中此功能,KILL 程序将结束运行,正常退出。

注意,在 KILL 程序运行过程中,可以随时使用 Esc 键,中止杀病毒过程,强行退出。

2) 超级巡捕 KV300

KV300 是新一代查解病毒的软件,具有开放式、广谱、智能、可扩充、自维护等功能。它具备开放式和封闭式两套查杀病毒方案,对付变形病毒有特效,而且对新病毒查出率高达 98%。支持网络环境和 Windows 平台。目前,世界上现有的几千种病毒,几乎没有能逃脱 KV300 的查解的。

KV300 的常用格式:

KV300 W

使用这种格式时,首先应该将带有写保护的原盘插入 A 或 B 驱动器中,启动时读一下原盘,成功后方可换成其他检查的软盘。用户调用 KV300 后,屏幕出现一主画面,根据主画面的菜单、功能键,再连续选择 A, B, C, ...盘扫描并且清除病毒。如果需要中途退出,使用 Esc 键即可。表 7.3 给出 KV300 功能键的作用。

表 7.3　KV300 调用后功能键作用

功能键	功　　能
F1	用 KV300 的第一套查病毒方式,即用外部开放扩展的病毒特征库和扫描过滤对所有文件和引导区进行全代码扫描搜索病毒。灵敏度和准确度极高,速度慢。
F2	用 KV300 第二套查病毒方式,即用程序内部封闭的另一套扫描方法,快速对此引导区和所有文件中的病毒进行扫描。速度较快。
F3	快速清杀已知病毒。
F4	用 KV300 的第一套查病毒方式,即用外部扩展的病毒特征库和扫描过滤法对引导区和 .COM, .EXE 文件进行全代码扫描搜索病毒。适应搜索网络服务器。
F5	对某一子目录内全部文件中的病毒进行扫描或清除。例:C:或 C:回车。
F6	可查看不属于 DOS 管理的硬盘隐含扇区,查看硬盘 0 面 0 柱 1 扇区主引记录及分区表,可在硬盘隐含扇区内查找被搬家的主引记录及分区表,并可向 A 盘备份保存。备份后,可用 KV300/ HDPT .DAT 的格式再恢复到硬盘 0 面 0 柱 1 扇区主引导区中,但事先应先用 KV300/ B 的格式将当前 0 面 0 柱 1 扇区主引导区备份到某一软盘,以防不对时,再原样恢复回去。
F7	显示病毒名及其基本性质,但事先应启动汉字系统。
F8	显示使用说明,但事先应启动汉字系统。
F9	显示版本号和简易说明等,但事先应启动汉字系统。
F10	自动测试和快速修复硬盘分区表。
Esc	任何状态下,按下此键可返回、终止、或退出。

功能键定义说明:

① F1, F4 定义使用开放式的病毒特征库 VIRUS .DAT 文件中病毒特征代码来扩展搜索病毒。

② F2 定义用常规的内部封闭式的查病毒原理和方法,快速查找已知病毒。

③ 启动 KV300 后的默认状态是 F3 = KILL,快速清杀病毒。

在清除病毒过程中,根据用户的不同要求,KV300 还有多种命令格式供选择。另外,用户应该常与开发者联系,以获得 KV300 的最新升级版本。

3) 高效杀病毒工具 NAV

NAV(Norton Anti Virus 3 .0)是目前世界上最先进的,集防毒、查毒和消毒功能于一体的综合性病毒防治软件,它能够对已知和未知的病毒进行识别和清除,可以安装在 DOS 和 Windows 环境下。

在 DOS 环境下,NAV 命令格式为:

NAV [驱动器名:][路径][操作参数]

其中参数　/ A 表示扫描 A:和 B:以外的所有驱动器;/ L 表示扫描 A:和 B:以外的所有本地驱动器;/ BOOT 表示只扫描检测特殊驱动器的引导扇区;/ MEM 表示只扫描内存;/ S 表示扫描指定路径;/ DELETE 表示删除被感染的文件。

4) 瑞星杀毒软件 RAV6 .0

计算机网络在带给人们先进通信手段的同时,也扮演了病毒传播的催化剂,病毒借助网络大大加快了其传染的速度。宏病毒就是专门感染 Windows 数据文件的恶性病毒,它寄生于结构极其复杂的 OLE(object linking embed)数据文件中。瑞星公司科研人员率先采用结构分离技术,成功地分析了 OLE 文件结构,研制出瑞星杀毒软件 RAN6 .0,实现了检测消除宏病毒而不损坏文件的重大突破。瑞星杀毒软件的使用与上述软件相似,但它有以下几个新特点:

· 简便的访问系统。瑞星电子公告牌 BBS 正式开通,任何计算机用户,只要具有 Modem和 Window 系统,就可以通过拨打(010)62641700 来访问关于反病毒的信息。

· 快捷的升级方法。使用瑞星的计算机用户通过 BBS,可以尽快实现反病毒软件的升级。

· 获取免费查毒软件。任何计算机用户,都可以通过 BBS 获取瑞星公司提供的查毒软件。用户可以直接到瑞星公司免费拷贝查毒软件,外地用户可以邮购。

5) 网络病毒的克星 Lan Desk Virus Protect

病毒克星 Lan Desk Virus Protect, 是 INTEL 公司与世界著名的电脑防病毒厂商趋势科技公司合作开发的网络防毒产品。它使网络管理员工减轻了防病毒的负担,保护网络系统免受染病毒的危险。它的主要功能包括:

· 持续扫描。它在 Netware 服务器层执行专门的监督程序,保持全天 24 小时监控,以实时作业方式扫描所有进出网络的文件。

· 超强侦毒。它不仅能清除一般病毒,而且能够查出隐匿行踪的变形变体病毒,并用统计方法找出病毒习性规则。

· 自动报告。当文件染病毒时,Lan Desk Virus Protect 会自动发出通知,然后据此采取处理行动。

· 记录档案。它能在一般时间内追踪网络病毒活动,使用户可以从 Lan Desk Virus Protect 档案中统计多种报表,其中包括扫描到的病毒类型、来源、时间,以及病毒活动摘要。

· 支持多种文件格式。Lan Desk Virus Protect 能在 Netware 网络上支持 DOS, Windows, OS 和 Macintosh 文件格式。

· 自动随时报警。一旦发现系统感染病毒,Lan Desk Virus Protect 立刻会向用户发出一个警报信号,根据用户设定方式自动进行扫毒。

· Lan Desk Virus Protect 不仅具有上述强大功能,而且使用方便,操作简单,完成整个程序的安装仅数分钟,清除病毒时也可根据用户的需要进行设定,并且当新的版本推出时,趋势公司会主动通知注册用户,免费更新升级。

6) 网络杀毒软件 Netkill

目前在网络环境中,服务器起着关键作用,由于服务器与客户机之间存在一一对应的关系,因此服务器极易感染上病毒,而且服务器感毒后,由于信息共享,可能会立刻将病毒传染网上所有的客户,给清毒工作造成困难。当前国内已建成的局域网中大多数使用 Novell 公司的 Netware 网络操作系统,中国金辰安全技术实业公司据此推出 Novell 版防治病毒软件包 Netkill。该软件包采用“扫描”与“清除”工作分离的方法,它在 Netware 服

务器上运行,实时监视服务器上文件的输入与输出,自动检测已知的病毒,一旦发现,就向网络管理员告警,然后进行病毒的清除工作,客户清除自己授权范围内的病毒文件,网络管理员清除整个服务器上病毒。下面概括一下 Netkill 主要特点:

· 操作简单,易于使用。Netkill 是针对不懂计算机专业的普通用户设计的,力求简单明了,安装方便,尽可能减少人机对话,最大限度地让软件自动完成检测和清除病毒的全过程。

· 简单通用的屏幕切换方式。Netkill 采用 Netware 通用的屏幕切换方式(Alt + Esc),可灵活的在主屏幕和病毒屏幕之间切换。若中途退出,按 Esc 键即可。

· 运行速度快。Netkill 软件包采用 C 语言和汇编语言混合设计,并对运行速度进行了优化,在客户机上扫描近 500 个子目录只需 6 分 30 秒,每秒可清除 16 个病毒文件。

· 实时检测病毒。Netkill 能真正做到实时检测服务器上文件进出的操作,当发现感染病毒的文件进出服务器时,屏幕上立刻显示出病毒的数目、染毒文件名和路径信息。

· 采用国际流行的系列号加密软件,降低了成本和安装难度,更加安全可靠。

· Netkill 升级方便,升级周期短。金辰公司不断将升级版本免费提供给登记用户。Internet 网址:http://www.cei.go.cn/homepages/sid/kj/c.htm 010-68557058。

以上是目前较为流行的防治病毒软件。但是,由于用户所使用的计算机系统不同,其系统感染病毒的种类繁杂,因此采取的杀毒手段和使用的软件产品也不完全相同。例如:MITAC 公司出的 ANTIVRUS.SYS 防毒程序,采用国际标准仿 Windows 界面制作的帝霸 DB95/DBNET、集成杀毒工具 CPAV/MSAV、冰岛 Fridrik Skulason 提供的反病毒共享软件包 F-PROT 等,都是很受欢迎的防治病毒软件,在计算机系统和网络的查毒、杀毒、防毒处理工作中发挥了重大作用。

7.4 典型病毒的危害与清除

迄今为止,发现的计算机病毒已经多达 7000 种,然而在我国出现频繁、危害较大的常见病毒一般有几十种。例如:大麻病毒、小球病毒、黑色星期五病毒、巴基斯坦病毒、米开朗基罗病毒、N64 病毒、生日快乐病毒、杨基多德病毒等。本节对典型病毒的危害进行了分析,并且概述了清除这些病毒的具体方法。

1. 大麻病毒

大麻病毒又称为“石头”病毒,或“STONE”病毒,它首次发现于新西兰的威灵顿,并很快流传,曾在我国泛滥一时,属于恶意性病毒。它主要感染使用 DOS 系统的微型计算机。

1) 破坏现象

该病毒感染硬盘主引导扇区,毁坏目录或文件分配表 FAT 信息。在系统启动时,它被自举程序装入内存 7C00H 处,并同时获得系统控制权,驻留内存的方法为常驻高端,占用 2K 字节,感毒后屏幕显示:“Your computer is new stoned !”或“Your PC is now stoned !”信息。

2) 清除方法

用 DOS 系统的 SYS 命令可以清除软盘上的大麻病毒;但最简捷的方法是用冰岛 Fridrik Skulason 的 F-PROT 程序或高效杀毒工具 NAV 软件直接清除。

2. 黑色星期五病毒

黑色星期五病毒又称"耶路撒冷"病毒、"疯狂拷贝"病毒、"希伯莱"病毒,它最早发现于以色列的希伯莱大学,由于传播速度快,隐藏深,行踪诡秘,在极短时间里造成全球扩散,属于恶性病毒。

1) 破坏现象

该病毒攻击所有的 .COM 文件和 .EXE 文件,使感毒后 .COM 文件增加 1808 字节,而 .EXE 文件则以每运行一次增加 1808 字节的速度,不断增长直到磁盘满为止。当系统运行感染黑色星期五病毒文件约 30 分钟后,屏幕左下方出现长方形亮块或作有规律的闪动,且伴有条形花纹,系统的运行速度成倍减慢直到无法进行下去。特别是当系统时间是 13 日又恰逢星期五时,任何可执行文件一运行就会被病毒删掉,从而造成大量文件的丢失。黑色星期五病毒的标识串"SUMSDOS"。

2) 清除方法

如果受感染的文件很多,那么可以用专门的杀毒软件清除,例如:Mcafee 的 Clean up,Fridrik SKulason 的 F-PROT 或 MicroCOM 的 Vierxpc 等杀毒软件。也可以删除全部受病毒感染的文件,重新用纯净的系统盘启动。

如果磁盘上只有个别文件被感染黑色星期五病毒,那么可以用 DUBUG 实用程序手工消除病毒:对于染毒的 .COM 文件,删除文件首部的病毒部分;对于染毒的 .EXE 文件,删除文件尾部的病毒部分,同时恢复文件的头部信息。

3. N64 病毒

N64 病毒又称为"Beijing"病毒或"June 4th"病毒,带有极强的政治目的,发现于我国 1989 年夏季,并流传于欧美和台湾。

1) 破坏现象

该病毒寄存于可执行文件 .EXE 和 .COM 文件中,一旦文件被系统运行,病毒即驻留内存高端,占用 2K 字节,并开始感染其后运行的所有可执行文件。染毒后系统运行速度变慢,经常引起死机。病毒发作时,中断正在执行的程序,在屏幕上出现用英文写的反动标语,造成计算机系统瘫痪。染毒后的 .COM 和 .EXE 文件结构如图 7.4 和 7.5 所示。

2) 清除方法

该病毒用诊断软件 CPAV 及 SCAN 不能检查出来,因此可根据病毒的传染机理进行检测:一是将可执行文件与原备份进行比较,若其长度比备份文件增加了 1831(病毒代码长度 1824 字节加上标识串长度 7 个字节)字节,则基本可以断定感染了此病毒;二是通过检查中断向量 1NT21 是否被修改。关于该病毒的清除可以用 KILL 杀毒软件完成,或者利用 DEBUG 命令人工清除。

病毒代码长度为 1824 字节
原 .COM 文件代码病毒对原代码无修改
病毒标识串长度为 7 字节 B8 00 02 05 6B F6 C7

图 7.4 .COM 文件染毒后的结构

原 .EXE 文件代码文件头部的参数被修改,代码长度被病毒按节取整
病毒代码长度为 1824 字节
病毒标识串长度为 7 字节 B8 00 02 05 6B F6 C7

图 7.5 .EXE 文件染毒后的结构

4. 米开朗基罗病毒

米开朗基罗病毒又称“米氏”病毒或“Michelangelo”病毒,该病毒发作日为 3 月 6 日,这一天是意大利著名画家米开朗基罗的生日,故得此名。

1) 破坏现象

米开朗基罗病毒感染主引导扇区和磁盘的引导扇区。当使用带有米氏病毒的磁盘启动系统时,该病毒用内存的随机数重写启动盘,从而破坏整个盘的数据信息。在引导型病毒中,这种破坏的危害最大、最彻底,而且其造成的损失无法恢复。米氏病毒是内存驻留的,占用内存高端 2K 字节空间,其病毒程序全长为 430 字节,病毒标识串为“E9AC00FS”。

2) 清除方法

发现米氏病毒后,简捷的方法是利用 KV300,McAfee 的 Clean UP 或 FridrikSkulason 的 F-PROT 杀毒软件自动处理,对系统和汇编语言熟悉的用户也可以用 DEBUG 命令人工清除。

5. 巴基斯坦病毒

巴基斯坦病毒又称为“Brain”病毒,或“Pakistani”病毒,是一个非常著名的引导型计算机病毒。

1) 破坏现象

巴基斯坦病毒由初始引导部分和传染表现两部分组成,初始引导部分在系统启动时进入系统,然后再装入其他传染部分。它从 FAT 表中查找空闲扇区,将 6 个扇区打上坏簇标记,用其中一个存放引导安装程序,其余 5 个存放该病毒的传染表现部分。该病毒侵占内存,占用 3K 到 7K 字节的存储空间,其早期版本仅感染软盘,并把被感染的卷标改为(C)Brain,感染硬盘。病毒的标识串为“1234”。

2) 清除方法

检测巴基斯坦病毒通过 CHKDSK 或 DIR 命令,若在显示器上看到“(C)Brain”卷标字样,或发现 3KB 坏块空间,说明感染此病毒,此外用 DEBUG 或 PCTOOLS 工具也可检测到。清除巴基斯坦病毒用杀毒软件中 NAV 或 Clean UP 都很有效。另外,利用 DOS 系统的 SYS 命令重写受感染的系统软盘或硬盘上的引导扇区,达到彻底清毒的目的。

通过本章的学习,我们了解到计算机病毒的基本知识,并重点掌握了防治病毒的实用方法,为今后安全高效地使用计算机打下良好基础。最后请记住:随时备份你的文件;使用合法正版软件;小心进入 Internet,下载后一定查毒;设置访问控制和认证系统;自己留有一份高效权威的杀毒软件,并及时升级更新。

附　录　一

中华人民共和国计算机信息系统安全保护条例

(1994 年 2 月 18 日中华人民共和国国务院令第 147 号)

第一章　总　　则

第一条　为了保护计算机信息系统的安全,促进计算机的应用和发展,保障社会主义现代化建设的顺利进行,制定本条例。

第二条　本条例所称的计算机信息系统,是指由计算机及其相关的和配套的设备、设施(含网络)构成的,按照一定的应用目标和规则对信息进行采集、加工、存储、检索等处理系统。

第三条　计算机信息系统的安全保护,应当保障计算机及其相关的设备、设施(含网络)的安全,运行环境的安全,保障信息的安全,保障计算机功能正常发挥,以维护计算机信息系统的安全运行。

第四条　计算机信息系统的安全保护工作,重点维护国家事务、经济建设、国防建设、尖端科学技术等重要领域的计算机信息系统的安全。

第五条　中华人民共和国境内的计算机信息系统的安全保护,适用本条例。

第六条　公安部主管全国计算机信息系统安全保护工作。

国家安全部、国家保密局和国务院其他有关部门,在国务院规定的职责范围内做好计算机系统安全保护的有关工作。

第七条　任何组织或个人,不得利用计算机信息系统从事危害国家利益、集体利益和公民合法利益的活动,不得危害计算机信息系统的安全。

第二章　安全保护制度

第八条　计算机信息系统的建设和应用,应当遵守法律、行政法规和国家其他有关规定。

第九条　计算机信息系统实行安全等级保护,安全等级的划分标准和安全等级保护的具体办法,由公安部会同有关部门制定。

第十条　计算机机房应当符合国家标准和国家有关规定。

在计算机机房附近施工,不得危害计算机信息系统安全。

第十一条　进行远程国际联网的计算机信息系统,由计算机信息系统的使用单位报省级以上人民政府公安机关备案。

第十二条　运输、携带、邮寄计算机信息媒体进出境的应当如实向海关申报。

第十三条　计算机信息系统的使用单位应当建立健全安全管理制度,负责本单位计

算机信息系统的安全工作。

第十四条　对计算机信息系统中发生的案件,有关使用单位应当在24小时内向当地县级以上人民政府公安机关报告。

第十五条　对计算机病毒和危害社会公共安全的其他有害数据的防治工作,由公安部归口管理。

第十六条　国家对计算机信息系统安全专用产品的销售实行许可证制度。具体办法公安部会同有关部门制定。

第三章　安 全 监 督

第十七条　公安机关对计算机信息系统保护工作行使下列监督权:

(一) 监督、检查、指导计算机信息系统安全保护工作;

(二) 查处危害计算机信息系统安全的违法犯罪案件;

(三) 履行计算机信息系统安全保护工作的其他监督职责。

第十八条　公安机关发现影响计算机信息系统安全的隐患时,应当及时通知使用单位采取安全保护措施。

第十九条　公安部在紧急情况下,可以就涉及计算机信息系统安全的特定事项发布专项通令。

第四章　法 律 责 任

第二十条　违反本条例的规定,有下列行为之一的,由公安机关处以警告或者停机整顿:

(一) 违反计算机信息系统安全等级保护制度,危害计算机信息系统安全的;

(二) 违反计算机信息系统国际联网备案制度的;

(三) 不按照规定时间报告计算机信息系统中发生的案件的;

(四) 接到公安机关要求改进安全状况的通知后,在限期内拒不改进的;

(五) 有危害计算机信息系统安全的其他行为的。

第二十一条　计算机机房不符合国家标准和国家其他有关规定的,或者在计算机机房附近施工,危害计算机信息系统安全的,由公安机关会同有关单位进行处理。

第二十二条　运输、携带、邮寄计算机信息媒体进出境,不如实向海关申报的,由海关依照《中华人民共和国海关法》和本条例以及其他有关法律、法规的规定处理。

第二十三条　故意输入计算机病毒以及其他有害数据危害计算机信息系统安全的,或者未经许可出售计算机信息系统安全专用产品的,由公安机关处以警告或者对个人处以5 000元以下的罚款、对单位处以15 000元以下的罚款;有违法所得的,除予以没收外,可以处以违法所得1至3倍的罚款。

第二十四条　违反本条例的规定,构成违反治安管理行为的,依照《中华人民共和国治安管理处罚条例》的有关规定处罚;构成犯罪的,依法追究刑事责任。

第二十五条　任何组织或者个人违反本条例的规定,给国家、集体或者他人财产造成损失的,应当依法承担民事责任。

第二十六条　当事人对公安机关依照本条例所作出的具体行政行为不服的,可以依法申请行政复议或者提起行政诉讼。

第二十七条　执行本条例的国家公务员利用职权,索取、收受贿赂或者有其他违法、失职行为,构成犯罪的,依法追究刑事责任;尚不构成犯罪的,给予行政处分。

第五章　附　则

第二十八　本条例下列用语的含义:

计算机病毒,是指编制或者在计算机程序中插入的破坏计算机功能或者毁坏数据,影响计算机使用,并能自我复制的一组计算机指令或者程序代码。

计算机信息系统安全专用产品,是指用于保护计算机信息系统安全的专用硬件和软件产品。

第二十九条　军队的计算机信息系统安全保护工作,按照军队的有关法规执行。

第三十条　公安部可以根据本条例制定实施办法。

第三十一条　本条例自发布之日起施行。

附录二　计算机信息系统保密管理暂行规定

(1998 年 2 月 26 日　国家保密局)

第一章　总　　则

第一条　为保护计算机信息系统处理的国家秘密安全,根据《中华人民共和国保守国家秘密法》,制定本规定。

第二条　本规定适用于采集、存储、处理、传递、输出国家秘密信息的计算机信息系统。

第三条　国家保密局主管全国计算机信息系统的保密工作。

各级保密部门和中央、国家机关保密工作机构主管本地区、本部门的计算机信息系统的保密工作。

第二章　涉 密 系 统

第四条　规划和建设计算机信息系统,应当同步规划落实相应的保密设施。

第五条　计算机信息系统的研制、安装和使用,必须符合保密要求。

第六条　计算机信息系统应当采取有效的保密措施,配置合格的保密专用设备,防泄密、防窃密。所采取的保密措施应与所处理信息的密级要求相一致。

第七条　计算机信息系统联网应当采取系统访问控制、数据保护和系统安全保密监控管理等技术措施。

第八条　计算机信息系统的访问应当按照权限控制,不得进行越权操作。未采取技术安全保密措施的数据库不得联网。

第三章　涉 密 信 息

第九条　涉密信息和数据必须按照保密规定进行采集、存储、处理、传递、使用和销毁。

第十条　计算机信息系统存储、处理、传递、输出的涉密信息要有相应的密级标识,密级标识不能与正文分离。

第十一条　国家秘密信息不得在与国际网络联网的计算机信息系统中存储、处理、传递。

第四章　涉 密 媒 体

第十二条　存储国家秘密信息的计算机媒体,应按所存储信息的最高密级标明密级,并按相应密级的文件进行管理。

存储在计算机信息系统内的国家秘密信息应当采取保护措施。

第十三条　存储过国家秘密信息的计算机媒体不能降低密级使用。不再使用的媒体应及时销毁。

第十四条　存储过国家秘密信息的计算机媒体的维修应保证所存储的国家秘密信息不被泄露。

第十五条　计算机信息系统打印输出的涉密文件,应当按相应密级的文件进行管理。

第五章　涉 密 场 所

第十六条　涉密信息处理场所应当按照国家的有关规定,与境外机构驻地、人员住所保持相应的安全距离。

第十七条　涉密信息处理场所应当根据涉密程度和有关规定设立控制区,未经管理机关批准无关人员不得进入。

第十八条　涉密信息处理场所应当定期或者根据需要进行保密技术检查。

第十九条　计算机信息系统应采取相应的防电磁信息泄漏的保密措施。

第二十条　计算机信息系统的其它物理安全要求应符合国家有关保密标准。

第六章　系 统 管 理

第二十一条　计算机信息系统的保密管理应实行领导负责制,由使用计算机信息系统的单位的主管领导负责本单位的计算机信息系统的保密工作,并指定有关机构和人员具体承办。

各单位的保密工作机构协助本单位的领导对计算机信息系统的保密工作进行指导、协调、监督和检查。

第二十二条　计算机信息系统的使用单位应根据系统所处理的信息涉密等级和重要性制订相应的管理制度。

第二十三条　各级保密部门应依照有关法规和标准对本地区的计算机信息系统进行保密技术检查。

第二十四条　计算机信息系统的系统安全保密管理人员应经过严格审查,定期进行考核,并保持相对稳定。

第二十五条　各单位保密工作机构应对计算机信息系统的工作人员进行上岗前的保密培训,并定期进行保密教育和检查。

第二十六条　任何单位和个人发现计算机信息系统泄密后,应及时采取补救措施,并按有关规定及时向上级报告。

第七章　奖　　惩

第二十七条　对在计算机信息系统保密工作中做出显著成绩的单位和人员应给予奖励。

第二十八条　违反本规定,由保密部门和保密机构责令其停止使用,限期整改,经保密部门、机构审查、验收合格后,方可使用。

第二十九条　违反本规定泄露国家秘密，依据《中华人民共和国保守国家秘密法》及其实施办法进行处理，并追究单位领导的责任。

第八章　附　　则

第三十条　军队的计算机信息系统保密工作按军队的有关规定执行。

第三十一条　本规定自发布之日起施行。

附录三　面向对象分布式系统 OZ 加密系统中的密钥类程序

程序 1　产生 DES 密钥对象类的程序

```
package JP.go.ipa.oz.system;
public class OzDESKey extends OzKey{
        protected String DESKey = null;
        public OzDESKey(String key){
            super(key);
            DESKey = key;
            }
            public String getDESKey(){
                return DESKey;
             }
}
```

程序 2　产生 DES 密钥类的程序

```
package JP.go.ipa.oz.system;

import java.util.Random;
import java.math.BigInteger;

public class DESKeyGengerator{
        protected String DESKey = new String();
        public OzDESKey genKey(){
            double current;
            byte k[] = new byte[64];
            for ( int i = 0; i < 64;i++ ){
                current = Math.random();
                k[i] = (byte)(Math.round(current * 9)/ 5);
                DESKey = DESKey.concat(String.valueof(k[i]));
            }
            return new OzDESKey(DESKey);
        }
        public String getDESKey(){
            return DESKey;
        }
}
```

程序 3　产生 RSA 密钥对象类的程序

```
package JP.go.ipa.oz.system;
  public class OzRSAKey extends Ozkey {
        proected String RSANumn = null;
        protected String RSAKey = null;
        public OzRSAkey(String numN, String key){
            RSANumN = numN; RSAKey = key;
        }
        public String getRSANumN(){
            return RSANumN;
        }
        public String getRSAKey(){
            return RSAKEY;
        }
  }
```

程序 4　产生 RSA 公开钥和秘密钥类的程序

```
package JP.go.ipa.oz.system;
import java.math.BigInteger;
import java.util.Random;
import java.lang.*;
import java.io.*;

public class RSAKeyGenerator{
    protected OzRSAKey pubkey = null;
    protected OzRSAKey prikey = null;
    protected int keylength;

    public OzRSAKey genPubKey(){
        return pubkey;
    }
    public OzRSAKey genPriKey(){
        return prikey;
    }
    public RSAKeyGenerator(int len) {
        keylength = len;
        genKey(keylength);
    }
    public RSAKeyGengerator(){
        keylength = 512;
        genkey(keylength);
    }
    private String bytetoString(byte[] b){
        String conv = "";
        int i;
        for(i = 0;i < b.length;i++){
            String str = new string(
                (((b[i]&128) > 0) ?"1":"0") + (((b[i]&64) > 0) ?"1":"0") +
```

```
                    (((b[i]&32)>0)?"1":"0")+(((b[i]&16)>0)?"1":"0")+
                    (((b[i]&8)>0)?"1":"0")+(((b[i]&4)>0)?"1":"0")+
                    (((b[i]&2)>0)?"1":"0")+(((b[i]&1)>0)?"1":"0")
                );
                conv+=str;
            }
            //System.out.println("\ni="+i);
            return conv;
        }
        private void genKey(int keylength){
            byte[] x1=new byte[2];
            BigInteger m,a,p,q,n,fy,e,d,t1,t2,ci,a1,p1,q1;
            int xnl,xel,xdl;
            byte[] x,xn,xe,xd;
            String numN,pubKeyE,priKeyD;
            Random current=new Random();
            m=new BigInteger(10,current);
            p=new BigInteger(512,100,current);
            q=new BigInteger(512,100,current);
            n=p.multiply(q);
            t1=p.subtract(m.valueof(1));
            t2=q.subtract(m.valueof(1));
            fy=t1.multiply(t2);
            do{
                d=new BigInteger(512,100,current);
                p1=d.gcd(fy);
                q1=m.valueof(1);
                System.out.print(".");
            }while(((p1.compareTo(q1)==0)&&(d.compareTo(p)==1)&&(d.compareTo
(q)==1)&&(d.compareTo(fy)==1))==false)
            e=d.modInverse(fy);
            xn=n.toByteArray();
            xe=e.toByteArray();
            xd=d.toByteArray();
            //System.out.println(bytetoString(xn)+"\n");
            //System.out.println(bytetoString(xe)+"\n");
            //System.out.println(bytetoString(xd));
            numN=bytetoString(xn);
            pubKeyE=bytetoString(xe);
            priKeyD=bytetoString(xd);
            pubkey=new OzRSAKey(numN,pubkeyE);
            pubkey=new OzRSAKey(numN,priKeyD);
    }
}
```

参 考 文 献

1 Diffie W . Hellman M E . New directions in cryptography . IEEE Transaction on Information Theory: 1976 . IT-22: 644～654

2 卢开澄,计算机密码学,北京: 清华大学出版社,1990

3 李向宇等 .计算机反病毒技术实用指南 .北京: 国防工业出版社,1992

4 Jia Jing et al . Realization of a Digit Signature System . In: Proc of the 6th China-Japan International Conference on Computer Application, 1994

5 王锡林等 .计算机信息系统安全与反病毒 .北京: 电子工业出版社,1995

6 Schnnier B . Applied Cryptography-Protocols, Algorithms and Source Code in C . 2nd Edition . New York: John Wiley & Sone . Inc, 1996

7 陈淑仪等 .EDI 技术 .北京: 人民邮电出版社,1996

8 张维明等 .信息系统建模技术与应用 .北京: 电子工业出版社 1997

9 Peter T .Daris & Barry D .Lewis . Computer Security For Dummies, 1997

10 袁忠良 .计算机病毒防治实用技术 .北京: 清华大学出版社,1998

11 王丽娜等 .多媒体环境下语音保密体制的研究与实现 .小型微型计算机,1998 .4