

现在我们看一看问题所在。Virgil在Word里建立了一个文档并创建了包含OPEN FILE功能的宏。这个宏含有一个宏病毒代码。然后他将文档发送给受害人，受害人很自然地打开文件（假设E-mail程序还没有打开文件），导致OPEN FILE宏开始运行。既然宏可以包含任意程序，它就可以做任何事情，如感染其他的Word文档，删除文件等。对Microsoft来说，Word在打开含有宏的文件时确实能给出警告，但大多数用户并不理解警告的含义并继续执行打开操作。而且，合法文件也会包含宏。还有很多程序甚至不给出警告，这样就更难以发现病毒了。

随着E-mail附件数量的增长，发送嵌有宏病毒的文档成为越来越严重的问题。比起把真正的引导扇区隐藏在坏块列表以及把病毒藏在中断向量里，这样的病毒更容易编写。这意味着更多缺乏专业知识的人都能制造病毒，从而降低了病毒产品的质量，给病毒制造者带来了坏名声。

8. 源代码病毒

寄生病毒和引导区病毒对操作系统平台有很高的依赖性；文件病毒的依赖性就小得多（Word运行在Windows和Macintosh上，但不是UNIX）。最具移植性的病毒是源代码病毒（source code virus）。请想象图9-27，若该病毒不是寻找可执行二进制文件，而是寻找C语言程序并加以改变，则仅仅改动一行即可（调用access）。infect过程可以在每个源程序文件头插入下面一行：

```
#include <virus.h>
```

还可以插入下面一行来激活病毒：

```
run_virus();
```

判断在什么地方插入需要对C程序代码进行分析，插入的地方必须能够允许合法的过程调用并不会成为无用代码（如插入在return语句后面）。插入在注释语句里也没什么效果，插入在循环语句里倒可能是个极好的选择。假设能够正确地插入对病毒代码的调用（如正好在main过程结束前，或在return语句结束前），当程序被编译时就会从virus.h处（虽然proj.h可能会引起更多的注意）获得病毒。

当程序运行时，病毒也被调用。病毒可以做任何操作，如查找并感染其他的C语言程序。一旦找到一个C语言程序，病毒就插入上面两行代码，但这样做仅对本地计算机有效，并且virus.h必须安放妥当。要使病毒对远程计算机也奏效，程序中必须包括所有的病毒源代码。这可以通过把源代码作为初始化后字符串来实现，特别是使用一串32位的十六进制整数来防止他人识破企图。字符串也许会很长，但是对于今天的大型代码而言，这是可以轻易实现的。

对初学读者来说，所有这些方法看起来都比较复杂。有人也许会怀疑这样做是否在操作上可行。事实上是可行的。Virgil是极为出色的程序员，而且他手头有许多空闲时间。读者可以看看当地的报纸就知道了。

9. 病毒如何传播

病毒的传播需要很多条件。让我们从最古典的方式谈起。Virgil编写了一个病毒，把它放进了自己的程序（或窃取来的程序）里，然后开始分发程序，如放入共享软件站点。最后，有人下载并运行了程序。这时有好几种可能。病毒可能开始感染硬盘里的大多数文件，其中有些文件被用户共享给了自己的朋友。病毒也可以试图感染硬盘的引导扇区。一旦引导扇区被感染，就很容易在核心态下放置内存驻留病毒。

现在，Virgil也可以利用其他更多的方式。可以用病毒程序来查看被感染的计算机是否连接在局域网上，如一台机器很可能属于某个公司或大学的。然后，就可以通过该局域网感染所有服务器上未被保护的文件。这种感染不会扩散到已被保护的文件，但是会让被感染的文件运行起来十分奇怪。于是，运行这类程序的用户会寻求系统管理员的帮助，系统管理员会亲自试验这些奇怪的文件，看看是怎么会事。如果系统管理员此时用超级用户登录，病毒会感染系统代码、设备驱动程序、操作系统和引导扇区。犯类似这样的一个错误，就会危及局域网上所有计算机的安全。

运行在局域网上的计算机通常有能力通过Internet或私人网络登录到远程计算机上，或者甚至有权无须登录就远程执行命令。这种能力为病毒提供了更多传播的机会。所以往往一个微小的错误就会感染整个公司。要避免这种情况，所有的公司应该制定统一的策略防止系统管理员犯错误。

另一种传播病毒的方法是在经常发布程序的USENET新闻组或网站上张贴已被感染病毒的程序。也可以建立一个需要特别的浏览器插件的网页，然后确保插件被病毒感染上。

还有一种攻击方式是把感染了病毒的文档通过E-mail方式或USENET新闻组方式发送给他人，这些