The Computer Virus Situation is not Encouraging

Belden Menkus

As recognition of the danger inherent in the spread of computer viruses continues to grow, reactions from some segments of the information systems security community are beginning to sound familiar. A wide variety of quick fixes to the problem are being offered, often by self-appointed and often misinformed experts on both the virus problem and means to remedy it.

Too little has been learned from the Hacker Experience

Some of these same individuals responded in much the same way to the early reports of hacker activities. In particular, it again is being contended that

1. This is not a significant problem; it will go away if the news media will quit talking about it. In the case of the hackers, news people eventually did turn their attention to other things. Unfortunately, the hacker problem did not go away; it quietly became progressively worse. There is no reason to believe that anything different will occur with respect to computer viruses. This phenomenon will not disappear readily.

The computer virus problem results from the refinement of a specialized form of industrial sabotage. It has taken a long time to manifest itself in its present form. (John von Neumann, pioneering computer theorist, first discussed the concept of a self-reproducing program in his 1949 paper on the Theory and Organization of Complicated Automata.) The ancestors of the viruses have been with us for some time; the first self-replicating programs had been created in computing research sites by 1960 [1].

Very powerful Trojan horses, of which the computer viruses are a form, were already disrupting major data processing facilities by the end of the 1960s. They have continued to reappear since then. For instance, at 7:30 AM on 11 April 1980 all of the IBM 4341s already delivered to customers stopped dead. Reportedly, a disgruntled IBM employee had placed a logic bomb, another form of Trojan horse, in the master clock used in manufacturing the unit [2].

There are no Simple Solutions

2. Whatever problems may be caused by computer viruses can be resolved with a "quick fix". Again, in the case of the hackers, much of the reaction to the growth in general awareness of their activities took the form of what might be termed miracle cure hardware and software products that proved, upon close examination, to be of limited effectiveness.

Attempts to respond to the computer virus threat have proven to be more complex. Among those most at risk, there is a willingness to seek comparatively inexpensive solutions to the problem by relying on a variety of freeware and shareware antivirus software offerings that are, at times, of
questionable value and integrity. In at least one instance, a computer virus has been planted in a shareware product designed to combat viruses [3]. An apparent lack of awareness among shareware producers of a potential exposure of their products to virus infestation apparently is reflected in a recently published guide to these products [4]. The article fails even to allude to this subject.

Commercially offered computer virus protection software products have proven in the main to be discouraging. Many do not work at all, or perform unreliably. In some instances these products have actually proven to be vaporware, they were not delivered after they were announced. In at least one instance, a call seeking more information about a press release describing one of these products disclosed that the developer's telephone number had been disconnected while the announcement had been moving through the mails.

Self-designated experts on computer viruses are proliferating. Unfortunately, a significant number of these people simply do not understand what they are talking about. Misinformation proliferates as these people are interviewed by reporters who do not understand either computing or information security processes. This situation is being confused further as various software products, originally meant to improve microcomputer user performance, are being offered to perform tasks for which they were not designed, i.e. virus detection or protection.

No Expectation of Innocence

(3) Those responsible for these incidents are comparatively innocent pranksters with relatively limited competency. They won't do anything more malicious or sophisticated. Hackers are by no means always innocent in their activities. One unidentified group of them, for instance, gained control in October 1987 of a voice message system operated by Certified Grocers of California, Limited, a Los-Angeles-based wholesaler. During the time that they controlled this network, the hackers were reported [5] to have used it to merchandise stolen credit card numbers, male and female prostitutes, and various types of drugs.

Hackers have also become progressively more sophisticated. One illustration of this change is a West German who during a 2 year period compromised more than 30 sensitive computing facilities with ties to the U.S. Defense establishment [6]. There seems to be no foreseeable limit to the sophistication and ingenuity displayed by hackers.

The developers of computer viruses are no less knowledgeable [7]. There already is evidence of a comparable growth — in a far shorter period — in the competency and sophistication of the developers of computer viruses. Those who are developing computer viruses appear to be concerned mainly with demonstrating their power to other computing professionals. Improvements detected in virus design include elimination of target file date changes and other evidences of infection, reduction in virus size, and enhancement of the means used to mask the virus' presence in an infected computer.

Computer virus designers do not appear, at present, to be concerned about the consequences of their actions; there is no reason to believe that they will end their activities in the near future. There is, instead, every reason to believe that the computer virus problem will continue to grow, create a variety of information integrity and control problems, and eventually force major changes in the ways in which both operating system and application software are structured.

The Problem is "Built-In"

One of the inherent vulnerabilities in current software design practices can be illustrated very easily. Ken Thompson of the Bell Laboratories is reported [8] to have demonstrated in 1984 a way in which the Unix compiler could be compromised so as to expose every user's files to apparently legitimate — but actually illegal — access. Thompson also
showed, according to this account, how a second modification of the compiler would permit the first modification to survive undetected.

Strategies for Virus Infection Limitation

A number of recommendations for what is being described as safe computing have been circulated widely. Largely, these injunctions call for caution before initiating the USC of software whose freedom from viruses cannot be assured. A number of broader-based strategies for limiting virus infection also have been proposed. For instance, it has been suggested that in the 386 microchip environment it will be possible to isolate a virus absolutely because a single microcomputer will be partitioned into numerous virtual machines. Another approach would be to issue program code in a non-modifiable CD-ROM form rather than on a conventional erasable magnetic diskette. Again, it might be feasible to issue some programs in the form of conventional ROM microchips. In all three instances it does not appear that these protective approaches can be realized on any broad basis before the early 1990s.

In the interim, several other actions would seem to be prudent. These include

(1) Designing trap interfaces to local area network gateways and other telecommunications ports that will prevent unmediated direct access from outside sources to disk content.

(2) Segmenting the interconnection of data-center-resident DASD — and the program that manages the use of this space — to reduce the possible impact of virus infection of mainframe-supporting memory.

A Consistent Lack of “Quality”

To be considered as possessing quality, programs designed to combat viruses should meet two tests. First, they should be suitable for use by someone engaged in computing in a conventional business environment. Secondly, they should facilitate establishing and maintaining a virus-free computer environment. Typically, currently available antivirus software products cannot be deemed to meet these tests for the following reasons.

(1) These products fail to conform to anticipated user requirements. In particular, a conceptual defect is reflected in this entire class of software. They do not prevent the infection of software by a virus. Rather they simply inform the system user after the infection has occurred. This is somewhat like being told after one has been hit by a large truck what the brand of fuel in its tank is.

Use of the term filter to describe programs like these runs counter to both its standard definition and the way in which its use commonly is understood. A filter is a mechanism that prevents a particular substance from moving from one space to another. This type of program does not perform that sort of preventive function. It should be described more correctly as an intrusion or infection detector.

(2) These products are difficult to use — and it is still true, as Calvin Moors noted [9] nearly 30 years ago, that people will use any system only when it is more painful not to use it. That is why, to draw a parallel, most United States automobile drivers continue not to use seat belts. They appear to consider the regular use of these devices to be a nuisance. Apparently, these people do not recognize the danger implicit in not using them. And, most who would use a microcomputer in the typical business environment would consider the use of most virus filter products to be a nuisance and would be willing to run the risk implicit in not using them.

(3) These products do not allow for possible advances in virus design — especially in terms of what might be described as improvements in virus complexity. In particular, the design of these antivirus products generally does not provide for the threat posed by the possible introduction of a virus embedded in a data file, rather than in a program.
B. Menkus/Computer Virus Situation is not Encouraging

These products tend to share a problem common to much of the microcomputer software currently available. They fail the first test that should be applied to any software product — in fact, to any system. Is this thing directed to satisfying the requirements of its potential user? If that person is defined, in this instance, as the average microcomputer user in a business environment, then it must be assumed that this individual

- is not oriented to computing as a job goal; it is meant to be a means to an end and nothing more;

- is more interested in the uses of the data than in the intricacies of the software that is used;

- wants software in general to be simple to install and easy-to-use, without adding operational overhead or otherwise interfering with the effective doing of this person's basic job;

- seeks protection against virus infection that provides a safe or more trustworthy environment, and nothing else. The concept of multiple levels of protection, reflected in the design of some of these products, is technically feasible, but operationally unrealistic in such an environment.

The typical anti-virus product does not appear to have been designed for the average user of a microcomputer in a business environment. These products appear to be intended for use by what might be described as the computer literate or, possibly better, the computer-obsessed individual.

An Illustration of the "Filter Problem"

One representative virus filter presently marketed in the United States further illustrates the deficiencies inherent in the currently available products of this type. This product appears to be neither significantly better — nor significantly worse — than most of its competitors.

This product is a checksum calculation process packaged to look like something else. Basically, the developers of this software offer little more than what might be described as cold comfort to the users of their product. The product may identify the presence of some types of viruses; it will not necessarily identify others. The product's developer appears to understand its limitations, but they are not presented in a clear fashion to the user. The documentation for the product does advise the user, however, almost as an aside, that whatever effectiveness the product may have is limited to those viruses that use the BIOS as the avenue of infection.

This product is weak structurally on several counts.

1. While the product is intended for use with hard disks and memory extension boards, no direction is given for recovering from disclosure of a virus infection in these devices. One puzzling aspect of the advice for recovering from infection on a diskette is the presentation of an option: Either destroy the physical diskette or send it to the product's developer for what is described as otherwise undefined analysis. It is not clear whether the developer is collecting viruses and their variants or if the infected diskettes are used in an effort to upgrade this anti-virus product.

2. A simple failure to save the list of protected files will abort the entire protection scheme. This is not stated clearly in the product's documentation.

3. Use of the checksum comparison leads to the periodic identification of legitimate changes in file content as viruses. Frequent false alarms in security system operation almost always lead to a pattern of ignoring routinely all warnings of security problems.

Another Approach Is Needed

Either some better type of software product needs to be developed to counter the threat of computer viruses or a different type of approach to maintaining the computing environment is needed. At this point, there do not appear to be any simple
remedies. The activation of powerful computer viruses appears to represent the end of the open computing environment envisioned by John von Neumann and other pioneers in data processing some 40 years ago.

References

---

Belden Menkus has been a full-time consultant to management since 1968. He is accredited by the Society of Professional Management Consultants (SPMC) and is a Certified Information Systems Auditor, a Certified Records Manager and a Certified Systems Professional. He writes and lectures extensively on various aspects of business management. He is executive editor of *Journal of Systems Management* and a regular contributor to *Software News*, *Administrative Management*, *EDP Auditor and Business Insurance*. In addition, he is a member of the editorial board of *Corporate Crime and Security*, and the editor of *Data Processing Auditing Report*.

He is a Fellow of both the British Institute of Administrative Management and the American Association of Criminology. He is a member of the Board of the EDP Quality Assurance Institute, and the Panel of Arbitrators of the American Arbitration Association, the Association for Systems Management, the New York Crime Prevention Council, and the Business Forms Management Association.

He has twice been awarded the silver medallion of the American Management Association and has received the distinguished service citations of both the National Micrographics Association and the Association of Records Executives and Administrators. He has been named a life honorary member of the Federal Emergency Management Administration Staff College faculty.