The Application of Epidemiology to Computer Viruses

W.H. Murray

Ernst and Whinney, 2000 National City Center, Cleveland, OH 44114, U.S.A.

Recently there have been a number of news reports on computer "viruses." This column is intended to give you an understanding of viruses and the issues that they raise.

Virus is Defined and Described

Cohen [3] defines a computer "virus" as a program that can infect other programs by modifying them to include a possibly evolved copy of itself. It is often a special case of a "Trojan Horse" (a malicious entity concealed inside a benign entity for the purpose of getting it through a protective boundary). It is distinguished by the facts that it is self-replicating and that it attaches itself to a host (program or other data object) for the purpose of concealing and transporting itself from one domain to another.

When executed, a virus causes a copy or copies of itself to be inserted in, or attached to, other programs or files. The purpose is to get these copies inserted into other domains. By so doing, the virus program can expand its influence. It can contaminate all of the compartments (user identities, virtual machines, memory spaces etc.) within a system or network of similar, or known and recognizable systems.

The name, "virus," is suggestive of the way in which the program spreads and infects things which it touches. However, the analogy holds so well that, as we shall see, it also suggests effective protective measures.

The Possible Behavior of a Virus is Considered

The virus may, like any Trojan Horse program, also do other things, benign or malicious. These might include such things as destroying data found in the infected domains. (While this is disruptive, proper back-up might have given protection. It should be noted that improper back-up may provide a place for the virus to hibernate and may cause reinfection.) It does not benefit the originator. However, if the originator is connected to the victim, then the virus might be able to do things that benefit the originator. For example, it might send copies of data that it finds in the infected domains back to the originator. Attacks have been demonstrated in which the virus could dupe a user into surrendering his ID and password to it so that these could be sent back to the originator. If the originator is so connected to the target domain that he could logon to it, then this information might permit him to do so.

Cohen [3] has demonstrated that in a closed system of several hundred users, that a virus might infect every user domain in less than a day. This might permit the originator of a properly designed virus to gain total control of the infected system.

The Potential Consequences are Considered

It should be clear that in a hospitable but sensitive or vulnerable

While this work is my own, I got the central idea from Fred Cohen [3].

Two dictionaries checked by the author fail to give a plural form of virus. My personal preference is for "viri." None the less, I will use the colloquial "viruses."

0167-4048/88/$3.50 © 1988, Elsevier Science Publishers Ltd.

environment, a virus could spread rapidly and do a great deal of damage. It is clear that hospitable environments exist. It is equally clear that sensitive and vulnerable environments exist. The successful spread of benign viruses in hospitable environments has been demonstrated [3]. Instances of disruptive, and even malicious, viruses have been reported [5]. To date, while disruptive to the victims, the effects have been tolerable to the community.

While the effect of any Trojan Horse on a particular system or user may be devastating to that system or user, it is likely to be tolerable to the community. It is the potential for the virus to spread in an epidemic manner, to so infect extended communities as to break down controls based upon separation of duties or compartmentation of privileges, that gives cause for public alarm.

Because of the benefits they bring, the system environments we are building will likely be more, rather than less, hospitable to the spread of viruses. That is, they will be more open, have larger user populations and more contact among the users. It is these characteristics that bring the benefits. For the same reasons we will be more dependent upon the systems.

Of course, every cooperative community is vulnerable to deviant behavior of only one or two members; the more cooperative, the more vulnerable. Most manage. A vulnerability does not equate to a problem. None the less, while negative effects are far from certain, the potential for them can only grow. It is useful for us to consider defenses and protective measures.

The Epidemiologic model

Analogies are triply instructive. The similarity between the analogy and that which it describes, may teach us things about the content of the problem that we might otherwise overlook. Second, the analogy makes the content easier to understand and communicate. Finally, at the point at which the analogy begins to fail, we learn yet a few things more.

The use of the name "virus", to refer to the kind of computer program we are dealing with, is suggestive of the way in which it spreads and how it behaves, that is to say, in the manner of an infectious disease.

The word epidemic comes from the Greek and means, literally, "upon the population." In epidemiologic medicine, disease is described as it occurs in a group or population, rather than as in clinical medicine, as it occurs in an individual. Epidemiology is that field of medicine concerned with the description of factors and conditions that are associated with the spread of an infectious process within a community. Since the behavior of a virus program is analogous that that of such an infectious process, the findings and strategies of this field may be useful to us [1].

Community, population, carrier, portal of entry, vector, symptom, modes of transmission, extra-host survival, immunity, susceptibility, sub-clinical, indicator, effective transfer rate, quarantine, isolation, infection, medium and culture are all terms from epidemiology that are useful in understanding and fighting computer viruses.

Symptoms of the Virus as Viewed by the Epidemiologist

When statistical epidemiology began in England in the 1830s [2], the only symptom that was reported and recorded with regularity was death. When correlated with location, this often pointed to unhealthy conditions. Today the epidemiologist is interested in other symptoms because they often provide evidence about the nature of the virus. Similar symptoms are often caused by viruses that share identifying characteristics and which yield to similar therapies.

The first noticeable symptoms of a computer virus may be a reduction from the expected in performance. One reported virus-like program had so filled the network with copies of itself, that it was not possible to get warnings to potential victims. Some viruses have been designed to call attention to themselves by displaying messages informing the victim about the damage. As with disease, the more destructive the symptoms, the sooner they will be noticed. A sudden increase in mortality signals a new disease. If the symptoms include flu-like symptoms, then it is more likely that the disease is spread in the same manner as flu. Widespread reports of data loss or application anomalies not only suggest a computer virus but some other indicators of its nature. For example, network overload suggests a virus that is spread via the network.

The Transmission of the Virus is Described

A virus is expelled (sneeze, SENDFILE) from an infected member (carrier or originator) of a community (family, users of a common system or network), on a vector (mucous, data object, file or program), through a medium (air, network or shared input–output devices or media) through a portal of entry (nose, network reader) to a target member of the community. Depending upon the susceptibility (e.g. immunity, similar language, command or instruction set) of the target and the satisfaction of necessary triggering conditions (passage of an incubation period, event on the system clock ,execution of the virus code) the subject may manifest symptoms (fever, pain, destruction or disclosure of files). Even where no symptoms, appear, the subject may manifest sub-clinical evidence of infection (give positive response for a test for the virus). Independent of such evidence of infection, the virus may still replicate or be expelled. Usually, these events will produce some evidence but it will often be overlooked or ignored.

Disease viruses are naturally inserted into the vector and the vector is naturally expelled into the environment. Program viruses are not usually so passive. Rather they insert themselves into the data object and cause the object to be placed on the medium. If the medium is the network, then it may address the vector directly to the next victim with its last victim indicated as the source.

If the population is small or isolated, then the number of individuals affected will be limited. If the amount of contact within the population is limited, then the spread of the infection will be slow.

Some Defenses are Considered

Hygiene, Prophylaxis and Antidotes

Obviously, the process can be interrupted at any of these points or steps. Parties manifesting symptoms or positive test responses can be isolated from the remainder of the community. A handkerchief, mask or other filter could be used. The system analogy is a filter program capable of recognizing and eliminating the virus, while passing other data objects.

Individuals can use mechanical or chemical measures to prevent being infected themselves or infecting others. The use of surgical gloves is an example. Since the appearance of computer viruses, there have appeared a number of prophylactic programs. These programs are used in the presence of risk factors such as the use of a new program from an untrusted source. They alter the behavior of the target environment so as to respond to potential virus-like behavior. For example, such a prophylactic program might take control when any attempt is made to put data on media and raise an alarm. If no such behavior was expected of the newly imported program, then corrective action is indicated. In other cases, sorting benign behavior from malicious behavior may be more difficult.

As with disease, the conscientious and proper use of prophylactics can be effective in protecting individuals and, to a lesser degree, the community. As with disease, there must be knowledge, motivation, availability, and timely use. As with disease, these are difficult to achieve.

Isolation

Individuals can avoid public places, high risk groups, or stay at home. System users can refuse to accept data objects from the network or to use shared devices or media. Of course, in the process, one gives up some of the benefits of living in a community. The individual trying to avoid the flu may give up company or affection; the user trying to avoid computer viruses gives up data programs. On the other hand, those who fail to exercise such hygiene, risk not only symptoms of their own but infecting many others.

Quarantine

In severe disease epidemics, public authorities can close schools, impose quarantines, or forbid public gatherings. System authorities can forbid the use of shared media or shut down the network. It should be noted that these are remedies to be used only in the presence of an infectious agent. They cannot be used as a default simply to avoid the potential. They are too destructive of the community.

Purges

Some viruses, such as those that thrive in the gastro-intestinal tract, can often be eliminated through the use of a purgative. It can be possible to eliminate a virus from a computer system by eliminating all data that can possibly be contaminated by the virus. In systems in which programs are stored separately, this purge can be limited to the program libraries. Since these are normally fairly stable anyway, the disruption of falling back to the last uncontaminated version or even primary sources, need not be as disruptive as if one had to fall back to an earlier version of the more active database. Of course, one must take care not to recontaminate the system in the process of applying the necessary updates to the new program library.

This also demonstrates the value of systems such as MVS which tend to isolate the storage used for programs from that used for data or the System/38 which employs "strongly typed data objects." These can be contrasted to systems such as CMS, Unix and PC-DOS which tend to store all data types intermingled.

In addition to being painful and disruptive, the purge only treats one victim at a time. Since the purge produces no immunity, it is possible for the victim to be reinfected. Therefore, to stem the epidemic, it may be necessary to administer the purge to all victims simultaneously.

Natural Immunity

Members of one species may not be vulnerable to the same viruses as another. Viruses are often target specific. Some viruses that produce symptoms or replicate in one species, may do neither in another. Thus a computer virus that has PCs as its target may not produce symptoms or replicate in a machine such as the 370 that has a different instruction set. Similarly, viruses that are written in BASIC will not run in the absence of the BASIC interpreter. Natural disease viruses have been said to mutate spontaneously, sometimes in response to a medication or a new environment. Computer viruses are not natural, they are artifacts. If they are not replicated as intended, they will usually die. However, they can become more virulent by design and intent.

In at least one case, a computer virus was introduced into a community that was much more extensive, sensitive or vulnerable than its maker envisioned. They have also been re-engineered in the face of remedies. Sometimes the re-engineering was done by the originator, sometimes by others, in the manner of a game.

Portal of Entry

While some diseases may enter the victim through a variety of portals, others are limited to one. The computer virus can only enter through one that will accept programs. As a rule, it cannot enter through ports which are limited to data. One need not fear that a hacker will insert a virus into the banking system via an ATM. Likewise, business application systems that segregate programming from use and programmer ports from user ports, may be infected through the limited number of programmer ports, but not through the more numerous user ports. Thus systems which limit the number of programming ports may be less vulnerable than those that permit programming at any port. In defending against viruses, one can focus one's efforts on the programming ports.

However, any port that can ever accept data that can possibly be named as a program, or affect an existing program, can be used to insert a virus. (Thompson [4] describes a Trojan Horse concealed in the source code of a compiler. Every time the compiler is recompiled to produce the object code for the compiler, a trap door is inserted in the new object.)

Effect of Incubation Period

Most natural disease viruses require some time to produce symptoms or reproduce in a new victim or to spread to yet others. This may give the clinician time to prevent their spread through therapy. Computer viruses may act so fast as to deny any time for an effective therapeutic response or they may lie dormant and undetected for a long period. While this may present an opportunity for effective response, it may produce a false sense of security or make it difficult to trace a virus back to its origin.

Investigative Epidemiology

When new disease symptoms begin to appear in the community, the epidemiologist begins observational investigation. He attempts to identify the characteristics that are shared by the victims. The purpose is to determine how the disease is being spread and thereby identify effective strategies for interrupting its spread.

When the clinician encounters a case of a virus, he treats it. The epidemiologist wants to know where it came from and where it is going, in the hope of identifying and isolating victims and instances of the virus. Computer systems or networks that keep a record of contacts between users can be useful in localizing instances of the virus. In some cases they may enable the authorities to identify the author of the virus. The potential for detection will deter some mischief.

Identification of the Pathogen

Having identified a group of victims of similar symptoms which appear to be spread in a particular way among members of a specific population, the epidemiologist will attempt to identify any microorganism that appears in all of those manifesting the symptoms, perhaps in other members of the risk population, but not in others. If one or more such organisms can be identified, the epidemiologist attempts to identify some causal relationship between the organism and the symptoms.

Once identified, the characteristics of the pathogen itself are studied to identify any vulnerabilities it may have that will assist in clinical treatment of the victim or epidemiologic treatment of the population. For example, if the pathogen cannot live in air or water, then that eliminates many possible manners of infection.

Once the pathogen has been identified, the epidemiologist examines possible vectors and mediums for its presence. If it appears in blood serum not in saliva or urine, that provides useful evidence about how the disease spreads and strategies for the control of its spread.

Once a computer virus has been identified, purging a particular instance of it from a particular victim will usually be straightforward. Once it has been identified, it will often be obvious from its content what vector and medium it employs for its spread.

Inspection for Viral "Tags"

The epidemiologist hopes that the pathogen has distinctive physical or chemical characteristics or indications that make it easy to detect and identify. Computer viruses are often given such easily identified tags as the author solves the problems of activation and vector insertion. Disease viruses, finding themselves in a new and susceptible host, will simply do their thing. A computer virus, on the other hand, like all Trojan Horse programs, must somehow become active or gain control of the system. That is, it must get itself executed.

Program Name

Most systems require that executable objects be named as such. MicroSoft's BASIC expects its programs to have a file name extension of BAS. Likewise MS–DOS expects command language files to have an extension of BAT, and program names to have extensions of EXE and commands of COM. Other systems may provide more choices but the list is still explicit and short. These naming conventions provide tags that can be used to recognize potential virus programs. They can be used to activate filters to prevent infection, to identify victims and to sterilize victims.

Not only may viruses be limited to the names of executable items, but they may be limited to names that are called. One technique is to use a bait name, one that appeals to the user. XMASCARD has been used effectively. Another technique, that works in environments in which different kinds of executable objects share the same name space, is to take on the name of a system command; LOGOFF has been used. Some systems, MS–DOS and VM/370 included, resist this by preferring their own commands to other objects with the same name, but only if they are in the same directory. Thus a program with the same name as a command but stored in a user directory might be called ahead of that command stored in a system directory. Even a program that changes its name to conceal its identity has a limited set to choose from. The limited set of useable names may be useful in identifying viruses, testing and filtering.

Vector Data Object

A virus usually employs an existing data object for its vector. In the process it will usually change the length of the vector. This difference in length may sometimes be helpful in distinguishing an infected vector.

Depending upon the effectiveness of the controls in the environment and its own sophistication, the virus may also alter the "creation date" or "date of last change" of the vector data object. Any variance between these and an expectation of them can also be useful in identifying contaminated objects.

In order to establish addressability to the vector data object, the virus will often contain its name. This name can also be a useful tag for identifying the virus.

Immunization

Once enough is known about a virus, it is often possible to create a vaccine. A vaccine is a powerful tool in the hands of the epidemiologist. When a large number of members of the community are immune, then the effective rate of transmission of the disease or virus is greatly reduced.

Computer viruses depend upon their knowledge of the behavior of their targets. They must know how to use the target in order to produce their symptoms, replicate and propagate themselves. Small changes in the behavior of a target, may keep the virus from working. For example, changing the name of the intended vector data object or changing the name of some command (e.g. SENDFILE) might keep the virus from working in a particular target. However, it should be noted that protecting a single target does not destroy the virus. The virus is all of its copies.

While the public often sees the existence of a vaccine as a panacea, the epidemiologist realizes that it is only a tool. Although the Smallpox vaccine was very effective within small isolated communities, it took more than 50 years to eliminate the virus. The epidemiologist also knows that a vaccine has risks of its own. Its use can be expensive, dangerous or disruptive. Its use is justified only in the presence of disease.

The System Manager as Epidemiologist

In the case of disease, we have institutionalized our response in the form of the World Health Organization, Department of Health, the National Institutes of Health and the Centers for Disease Control. With an occurrence of a computer virus, the response will be from the system manager. The system manager will have to play the role of epidemiologist.

First he must encourage good computer hygiene and prophylaxis and must actively discourage such dangerous behavior as accepting, using or sharing programs or data from unknown or untrusted sources.

The system manager should be alert to any evidence of the presence of viruses. Symptoms of viruses such as network, system or application anomalies should be promptly investigated.

Finally, he must have a plan for dealing with any viruses that appear. Since a virus may never appear and since both the virus and the defenses against it represent threats to system reliability and availability, such planning will normally be done as part of business continuity or contingency planning. It should reflect the potential for a viral attack, and identify strategies and tactics for dealing with it. While the strategies will likely be specific to your own applications and environment, tactics will include purging the system, disconnecting from sources of contamination, or employing filters or antidote programs. Some tactics may require the availability of specialized resources (e.g. a trained system programmer); the plan should identify sources for any such resources. If nothing else, the plan should assign the responsibility for the key decisions that must be made. Since the actions that are indicated may be very disruptive, this responsibility must be assigned to an executive with sufficient authority and discretion for those actions.

Conclusions

The exposure to computer viruses arises from the desire to share programs and other data. It arises from the desire to communicate, cooperate and coordinate. In short, it arises from the very human desire to live in a community.

There is a sense in which the vulnerability arises from the credulity of computer users. There is another in which it arises from the obscurity of the codes in which they communicate intent. However, it does not arise from any inherent weakness in computer security, any more than the vulnerability to propaganda arises from any inherent weakness in printing presses.

Every cooperative community is vulnerable to the disruptive effects of a lie. Every interdependent community is vulnerable to the deviant behavior of a few individuals.

All systems are vulnerable to Trojan Horse attacks. Most attacks depend, at least in small part, on the cooperation of the victim. At a minimum, the victim must accept a data object from an unreliable source. At some cost, most victims can protect themselves.

A virus is a special case. It appears to have come from a known or trusted source. The success of its attack does not require that all victims, or even any particular victim, be duped or cooperate, but only a sample. Its effect is not necessarily limited to a single user. Finally, it is capable of so infecting a system or network as to compromise controls based upon separation of duties or compartmented privilege.

This paper has considered the spread of the virus in the community and the defenses against that, rather than the treatment of the symptoms that might appear in a particular victim. It has drawn on the science of medical epidemiology to identify effective strategies for limiting the spread and effect of these viruses.

The author recognizes that, as in biology, there are limits to the effectiveness of all these measures. The limits are in part in the nature of viruses and epidemics, in part in the fact that computer viruses are designed for their targets. A determined attacker could design for, or respond to, many of the measures outlined here. Of course, there is an analog for this in epidemiology: the mutating or resistant virus. The pathogen responds to the medicine. The system managers detect the virus and take action to thwart it. The attacker learns of these and responds. The system manager detects the changes and responds again.

There is no panacea. There does not appear to be a general vaccine that is likely to be effective against all viruses in all environments for all time. There is no general defense against the effects of bad data.

Good hygiene helps, but it will not give absolute protection. It is not possible to protect all users or compartments. However, proper response based upon the epidemiological model and the measures that it suggests, is likely to be at least partially effective against particular instances of a virus. Most of these measures are so disruptive that they should be used only in the face of a known attack.

A Personal Word

All this having been said, I am sanguine. God is still in his heaven. The environment is generally benign. The community is resilient. Most individuals are acceptably polite, orderly and well behaved. On the list of vulnerabilities in our complex society, this one is distinguished primarily by its novelty. Unlike some of the more intractable ones, this one will yield to good will. In the face of genuine evil intent, I prefer it to plastic explosives in power plants.

References

[1] *The Encyclopaedia Britannica*, pp. 640–643.

[2] J. Burke, *The Day the Universe Changed*, Little, Brown and Company, Boston, pp. 193–237.

[3] F. Cohen, *Computer Viruses—Theory and Experiments, DoD/NBS 7th Conf. on Computer Security*, originally appearing in IFIP-Sec 84, also appearing in *Comput. Secur.*, 6 (1987) 22–35, and other publications in several languages.

[4] K. Thompson, *Reflections on Trusting Trust, Turing Award Lecture, 1984*, CACM, August 1984.

[5] *Virus Hits Computer Programs*, The Detroit News/Associated Press, January 11, 1988.

Anatomy of a Virus Attack

Dr. Harold Joseph Highland, FICS

Almost everyone has heard the apocryphal story of a virus wiping out a hard disk and displaying the message: "Arf! Arf! I Got You!" Late in 1985 we accidentally infected the hard disk of one of our microcomputers while testing a virus detection program. Because we were certain that the virus was not present before we executed the virus filter program, it was simple to remove the virus from our system.

(a) We had a complete set of back-ups that were made the day before testing the virus filter.

(b) An emergency program was used to overwrite our hard disk with zeroes and ones. The system was shut off for 5 minutes to make certain that the virus did not remain in memory.

(c) We turned on the power and used a special boot disk in drive A. (The emergence 0–1 program is part of a special boot disk that is kept in a unit on the right side of the microcomputer.)

(d) We reformatted the hard disk and used the back-up disks to restore the files and programs on the hard disk.

The entire procedure took somewhat over 1 hour. It would have been more complex except that we were certain that there had been no prior computer virus infection. Nevertheless, as a result of the incident we promulgated Rule #1: "*All virus testing must be done only on a two-floppy disk system.*"

Preparing the Test Disk

Having collected a series of viruses over the past few years, we decided to examine the progress of a virus infection. The special bootable test disk for this purpose had

(1) four executable programs, two .COM and two .EXE programs, and

(2) four virus programs from our collection, two .COM and two .EXE programs.

The directory of the test disk when it was prepared is shown in Fig. 1. Naturally, the operating system files, IBMBIO.COM (16 369 bytes) and IBMDOS.COM (28 477 bytes), were not printed in the directory because they are hidden files. Also shown in Fig. 1 is a map of the entire disk which was obtained