The Internet Worm Incident

Eugene H. Spafford
Department of Computer Sciences
Purdue University
West Lafayette, IN USA 47907-2004
spaf@cs.purdue.edu

On the evening of 2 November 1988, someone "infected" the Internet with a worm program. That program exploited flaws in utility programs in systems based on BSD-derived versions of UNIX. The flaws allowed the program to break into those machines and copy itself, thus infecting those systems. This program eventually spread to thousands of machines, and disrupted normal activities and Internet connectivity for many days.

This paper explains why this program was a worm (as opposed to a virus), and provides a brief chronology of both the spread and eradication of the program. That is followed by discussion of some specific issues raised by the community's reaction and subsequent discussion of the event. Included are some interesting lessons learned from the incident.

1. Introduction

Worldwide, over 60,000 computers\(^\dagger\) in interconnecting networks communicate using a common set of protocols—the Internet Protocols (IP).[7, 15] On the evening of 2 November 1988 this network (the Internet) came under attack from within. Sometime after 5 PM EST, a program was executed on one or more of these hosts. That program collected host, network, and user information, then used that information to establish network connections and break into other machines using flaws present in those systems' software. After breaking in, the program would replicate itself and the replica would attempt to infect other systems in the same manner. Although the program would only infect Sun Microsystems Sun 3 systems, and VAX\(^\text{TM}\) computers running variants of 4 BSD\(^\circ\) UNIX,\(^\circ\) the program spread quickly, as did the confusion and consternation of system administrators and users as they discovered that their systems had been invaded. Although UNIX has long been known to have some security weaknesses (cf. [22], [13, 21, 30]), especially in its usual mode of operation in open research environments, the scope of the break-ins nonetheless came as a great surprise to almost everyone.

---

\(^*\) The presentation in [26] is a condensation of an early version of this paper.

\(^\dagger\) As presented by Mark Lottor at the October 1988 Internet Engineering Task Force (IETF) meeting in Ann Arbor, MI.

\(^\circ\) BSD is an acronym for Berkeley Software Distribution.

\(^\circ\) UNIX is a registered trademark of AT&T Laboratories.

\(^\text{TM}\) VAX is a trademark of Digital Equipment Corporation.
The program was mysterious to users at sites where it appeared. Unusual files were left in the scratch (/usr/tmp) directories of some machines, and strange messages appeared in the log files of some of the utilities, such as the sendmail mail handling agent. The most noticeable effect, however, was that systems became more and more loaded with running processes as they became repeatedly infected. As time went on, some of these machines became so loaded that they were unable to continue any processing; some machines failed completely when their swap space or process tables were exhausted.

By early Thursday morning, November 3, personnel at the University of California at Berkeley and Massachusetts Institute of Technology had “captured” copies of the program and began to analyze it. People at other sites also began to study the program and were developing methods of eradicating it. A common fear was that the program was somehow tampering with system resources in a way that could not be readily detected—that while a cure was being sought, system files were being altered or information destroyed. By 5 AM EST Thursday morning, less than 12 hours after the program was first discovered on the network, the Computer Systems Research Group at Berkeley had developed an interim set of steps to halt its spread. This included a preliminary patch to the sendmail mail agent, and the suggestion to rename one or both of the C compiler and loader to prevent their use. These suggestions were published in mailing lists and on the Usenet network news system, although their spread was hampered by systems disconnected from the Internet in an attempt to “quarantine” them.

By about 9 PM EST Thursday, another simple, effective method of stopping the invading program, without altering system utilities, was discovered at Purdue and also widely published. Software patches were posted by the Berkeley group at the same time to mend all the flaws that enabled the program to invade systems. All that remained was to analyze the code that caused the problems and discover who had unleashed the worm—and why. In the weeks that followed, other well-publicized computer break-ins occurred and many debates began about how to deal with the individuals staging these break-ins, who is responsible for security and software updates, and the future roles of networks and security. The conclusion of these discussions may be some time in coming because of the complexity of the topics, but the ongoing debate should be of interest to computer professionals everywhere. A few of those issues are summarized later.

After a brief discussion of why the November 2nd program has been called a worm, this paper describes how the program worked. This is followed by a chronology of the spread and eradication of the Worm, and concludes with some observations and remarks about the community’s reaction to the whole incident, as well as some remarks about potential consequences for the author of the Worm.

2. Terminology

There seems to be considerable variation in the names applied to the program described here. Many people have used the term worm instead of virus based on its behavior. Members of the press have used the term virus, possibly because their experience to date has been only with that form of security problem. This usage has been reinforced by quotes from computer managers and programmers also unfamiliar with the difference. For purposes of clarifying the terminology, let me define the difference between these two terms and give some citations as to their origins; these same definitions were recently given in [9]: