Lessons from the Code Red Worm

by M. E. Kabay, PhD, CISSP
Associate Professor, Computer Information Systems
Norwich University, Northfield VT

On June 19, 2001, the Computer Emergency Response Team Coordination Center (CERT-CC) issued Advisory CA-2001-13 <http://www.cert.org/advisories/CA-2001-13.html> warning of a buffer overflow in the Microsoft Internet Information Server software versions 4.0 or 5.0 running under Windows 2000 and beta-test versions of Windows XP. This vulnerability allows execution of arbitrary code on a susceptible machine; i.e., anyone can execute any instructions they like on an unpatched system.

The Advisory urged, “Since specific technical details on how to create an exploit are publicly available for this vulnerability, system administrators should apply fixes or workarounds on affected systems as soon as possible.”

One month later, on July 19, Advisory CA-2001-19 <http://www.cert.org/advisories/CA-2001-19.html> was issued announcing that the “Code Red” worm (a free-standing, self-propagating program that spreads through network connections) was exploiting the vulnerability announced in CA-2001-13. The best description of this worm’s internals that I have seen is from the Internet Security Systems X-Force <http://xforce.iss.net/alerts/advise89.php>. The worm’s functions are in three phases: scanning and propagation; flooding; and sleep. In the first phase, which is limited to the infected systems’ definition of the 1st to the 19th of any month, the worm code, which is memory-resident at all times until a system is rebooted (i.e., there is no disk file for re-loading the code), scans either a fixed list of pseudo-random IP addresses or (in more recent variants) a variable list of generated IP addresses. The initial variant’s repeated scanning of fixed addresses permitted researchers to monitor how many systems were infected; most agree that around 250,000 systems were infected within nine hours of the initial reports. Vulnerable systems are then infected and the process continues, causing a noticeable (albeit so far modest) rise in total Internet bandwidth consumption. During this phase, the original version of the worm substituted the message “Welcome to http://www.worm.com! Hacked by Chinese!” to all requests for an HTTP connection.

The flooding phase, from the 20th to the 27th of the month, originally targeted the numerical IP address for http://www.whitehouse.gov, which eluded the distributed denial-of-service attack by changing its IP address in the Domain Name System. However, it is clear that later variants of this worm can easily be modified to use other methods for targeting victims; in addition to using alphanumeric addresses, the worm could take a leaf from several recent viruses by looking for the target address in specific off-shore locations on the Net where law enforcement and technical collaboration are slower than in the US and other developed countries.

The worm then goes to sleep and does not appear to wake up again; later variants, however, could easily be programmed to reawaken.

The variants of the worm described at the time of this writing (end of July) were more adept at spreading (because they didn’t use the same list of IP addresses over and over) and they hid themselves from detection by not putting up the substitute message, allowing for a longer propagation phase without interruption.

A free scanner for this worm is available at <http://eeye.com/html/Research/Tools/codered.html>
There are some obvious lessons from this outbreak:
1) It took only a month from public discovery and patch of a vulnerability to an outbreak of an exploit.
2) It has taken more than a month for many administrators of vulnerable systems to apply the patch.
3) Variants of the worm appeared almost immediately, and they were worse than the first ones.
4) The number of unpatched systems is so high that even a simple attack can measurably affect Internet traffic and increase response time for Web connections.
5) All unpatched systems will continue to be vulnerable to this type of exploit.
6) The fundamental flaw that allowed for this attack is poor programming: buffer overflows imply that input strings are not being checked for length or otherwise edited, allowing strings to be interpreted as instructions. Manufacturers need to improve their quality assurance.
7) The originator of the attack may never be known.
8) The criminal hacker subculture has bred a group of people whose enjoyment of harm approaches the level of clinical sociopathy.
10) We are very close to major damage to the information infrastructure through self-propagating code that exploits the inability and unwillingness of management to support network administrators in keeping their system patches up to date.

Perhaps it would be interesting to see what would happen if the sociopaths of the criminal underground were confronted with significant rewards for turning in the perpetrators of this kind of exploit. Maybe it would also be interesting to see what would happen if a class-action lawsuit were launched against manufacturers who distribute fatally flawed code to millions of sites.

***

M. E. Kabay, PhD, CISSP is Associate Professor in the Department of Computer Information Systems at Norwich University in Northfield, VT. Mich can be reached by e-mail at <mkabay@compuserve.com>. He invites inquiries about his information security and operations management courses and consulting services.

Copyright © 2001 M. E. Kabay. All rights reserved.

Permission is hereby granted to *Network World* to distribute this article at will, to post it without limit on any Web site, and to republish it in any way they see fit.