Powerful system utilities that bypass normal controls can be used to damage data and code. Network managers can control such “superzap” programs by limiting access to them; software designers can help network managers by enforcing capability checking at run-time.

Security systems using menus can restrict users to specific tasks; the usual security matrix can prevent unauthorized access to powerful utility programs. Some programs themselves can check to see that prospective users actually have appropriate capabilities (e.g., _root_ access). Ad hoc query programs can sometimes be restricted to read-only in any given database.

On some systems, access control lists (ACLs) permit explicit inclusion of user sets which may access a file (including superzap programs) for read and write operations.

Aside from using normal operating system security, one can also disable programs temporarily in ways which interfere with (they don't preclude) unauthorized access; e.g., a system manager can reversibly remove the capabilities allowing interactive or batch execution from dangerous programs.

It may be desirable to eliminate certain tools altogether from general availability. For example, special diagnostic utilities which replace the operating system should routinely be inaccessible to unauthorized personnel. Such diagnostic tools could be kept in a safe, for example, with written authorization required for access. In an emergency, the combination to the safe might be obtained from a sealed, signed envelope which would betray its having been opened. I can even imagine a cartoon showing a sealed glass box containing such an envelope on the computer room wall with the words, "IN CASE OF EMERGENCY, BREAK GLASS" to be sure that the emergency crew could get the disk or cartridge if it had to.

When printing important files such as runs of checks, it may be wise to print "hot" instead of spooling the output. That is, have the program generating the check images control a secured printer directly rather than passing through the usual buffers. Make sure that the printer is in a locked room. Arrange to have at least two employees watching the print run. If a paper jam requires the run to be started again, arrange for appropriate parameters to be passed to prevent printing duplicates of checks already produced.

Regardless of all the access-control methods described above, if an authorized user wishes to misuse a superzap program, there is only one way to prevent it: teamwork. By insisting that all use of superzaps be done with at least two members of the staff present, one can reduce the likelihood of abuse. Reduce, not eliminate: there is always the possibility of collusion. Nonetheless, if only a few percent (say, two percent for the sake of the argument) of all employees are potential crooks, then the probability of getting two crooks on the same assignment by chance alone is about 0.04%. True, the crooks may cluster together preferentially, but in any case, having two people using privileged-mode DEBUG to fix data in a database seems better than having just one.
One method that will certainly NOT work is the ignorance-is-bliss approach. I have personally heard many network managers dismiss security concerns by saying, "Oh, no one here knows enough to do that." This is a short-sighted attitude, since almost everything described above is fully documented in vendor and contributed software library publications. Recalling that managers are liable for failures to protect corporate assets, I urge all network managers to think seriously about these and other security issues rather than leaving them to chance and the supposed ignorance of a user and programmer population.

* * *


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. Visit his Web site at <http://www.mekabay.com/index.htm> for papers and course materials on information technology, security and management.

Copyright © 2002 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.