Software Protection

Free enterprise will, naturally, cause us to look for effective ways to assure that we'll be rewarded for our efforts. An ideal method yields a "win — win" result for both developer and the customer. You are encouraged to explore the matter of software protection as one method for approaching this ideal. Before getting into specifics, however, let's first describe copy protection, a term bandied about for many years, so that we are able to clearly distinguish it from software protection.

Copy Protection

Nearly everyone who uses a personal computer has heard the term "copy protection" as it relates to software programs. In some product ads there are actual notations, NCP for No Copy Protection and CP for Copy Protection, that appear next to each product's description. This helps to inform prospective buyers of each software producer's stance on this matter.

Generally speaking, copy protection can be defined as any method which seeks to prevent duplication (or at least require extra effort to circumvent the protection scheme) of the software vendor's original program diskettes.

Earlier versions of some very popular products, such as dBASE and Lotus 1-2-3, used copy protection. Some vendors employ one or more methods of copy protection schemes in their software products. Among the techniques used are non-standard track formatting of floppies, use of laser beam generated holes in the floppy disk medium, and sophisticated install/deinstall programs for hard-drive installations which might create hidden files and subdirectories.

Oftentimes, the user would be required to keep the special copy protection "key disk" in drive A for execution of the vendor's program to take place. Pity the user whose key disk became unusable or whose hard drive was subjected to disk optimization programs which may somehow have scrambled the copy protection information.

At best, copy protection seems to represent a "win-lose" situation from a vendor-customer viewpoint. For any number of reasons, as well as the inconvenience frequently associated with copy protection schemes, these copy-protected products were regularly criticized by end users. In fact, pressure to eliminate copy protection forced many companies to drop it from later software versions or subsequent product releases.

Software Protection

Getting back to the "win-win" situation mentioned above, we'll focus on software protection. As used here, software protection refers to a method for controlling usage of a developer's software product by an end user. Specifically, a hardware device must be present on the particular system where the software program is to be executed.

In contrast to copy protection methods, software protection places no restrictions on the making of backup copies of the software program or on having the software installed on any number of systems.

The physical device serves the purpose of restricting operation of the associated software package to just that system where the device is connected. In essence, program functionality moves to whichever system has the device attached.

Usually, the device is connected to an already present IBM/Centronics parallel printer port, with the printer cable plugged into the other side of the device. Some vendor's devices can be used on serial ports, some on Small Computer System Interface (SCSI) bus. To maintain user transparency, the device should not alter or influence any operations involving the port or bus where the device is connected.

When Does Software Protection Make Sense?

If you plan to offer a moderately expensive software program and maintain control of its distribution, you should consider implementing software protection. Because a typical device costs less than forty dollars, you can quickly offset that expense. If someone wants to use more than one copy of your software program, you can simply supply him with another device, charging him according to whatever method and for whatever amount is appropriate.

At best, copy protection seems to represent a "win-lose" situation from a vendor-customer viewpoint.

Cloning

Some software developers even encourage cloning of their software as a form of cost-free advertising. They design their program to work in some limited fashion — perhaps allowing for all operations except saving of created files, or to run for only so long, or to allow only certain modules to operate — when the external device is not present. Later, once the tryout by the prospective customer is judged to be what is desired, the party could contact you, and you could take their order, shipping them a complete software package along with the software protection device.

Some Sources For Protection Devices

Software Security, Inc.
1071 High Ridge Road Stamford, CT 06905 (203) 329-8870, (800) 333-0407

Rainbow Technologies, Inc.
9292 Jeronimo Road Irvine, CA 92718 (714) 454-2100, (800) 852-8569

ProTech Marketing, Inc.
9600-J Southern Pine Blvd. Charlotte, NC 28217, (800) 843-0413

Which company you choose as a supplier will depend upon what matters you judge most important to your company's interests. You should consider such factors as price, availability, minimum order quantity, delivery schedules, accessibility to technical support (phone and Bulletin Board System), demo disk with test routines, hardcopy documentation, and whether or not device source code is provided so that you can create your own customized interrogation routines.

Implementing the Protection

During the actual program development phase of a software program, you may find that the matter of incorporating software protection is assigned a relatively low priority. You might feel that you want to get the program working before trying to do anything else with it. That's okay. However, if you are going about this for the first time, just be sure to allow for some "hooks" either in your overall program or in specific modules for subsequent inclusion of appropriate device interrogation routines.

Most suppliers of software protection devices offer an evaluation kit (a demonstration disk, a device, and perhaps a manual) which you can purchase. Some suppliers provide a great deal of information to significantly reduce your required effort for implementation. So, don't be afraid to ask questions as to just what you get and what things cost, and you should do all right.

At the simplest level, you want to have a 'yes' or 'no' answer to the question, "Is my device connected to this particular system, or isn't it?" Particulars of device implementation are not described here. You must, of course, evolve your own approach based upon a specific device selection.

At advanced levels, you may want to incorporate quite sophisticated methods of program encryption/decryption, checksums, and random interrogation sequences and processing as ways to discourage any hacker from trying to defeat your software's protection mechanisms.

If you create software programs for which you wish to protect your investment, consider incorporating an external hardware device as an integral part of your software package. Using this approach, you can control software execution, access to particular modules or features, or other operations, while maintaining transparent operation from the user's perspective. ME

Raymond Sonoff is president of Sonoff Consulting Services, Inc., P.O. Box 2027, Darien, CT, offering consulting, software development, and documentation services including high- and low-level language interfaces, database programs, MSIDOS utilities, and desktop publishing. Raymond can be reached at (203) 656-1518.