Microcomputer Software Piracy and Prevention Strategies

TOM SINGLETON
MILAM AIKEN
ASHRAF SHIRANI

University of Mississippi

Software piracy has lead to huge losses for software publishers, and yet they seem to be unable to find a solution to the problem. Managers in organizations using computer software must also be aware of the problem of software piracy because of their potential legal liability. The purpose of this paper is to demonstrate the cost of piracy to software publishers and organizations that use illegal copies of software. In addition, the paper examines current alternatives available to discourage the problem and emphasizes the roll of user education as a potential solution.

Software piracy (the unauthorized copying or duplication of proprietary software) is a major problem to the microcomputer world (Simpson and Thorn, 1989; Straub and Collins, 1990). Yet recently, piracy seems to have been forgotten as computing journals have published very few papers about the topic as more interest is paid to computer viruses and security from hackers (Simpson and Thorn, 1989). However, the problem of software piracy is of concern not only to software manufacturers and distributors, but also to software users and their managers because of their potential legal liability. Managers and users should also be concerned about the problem because a reduction in piracy may decrease the cost while increasing the quality of the software they purchase (Maude and Maude, 1984).

A review of the literature and a survey of 13 software publishers provide data on the costs of software piracy and methods that are currently being used to combat the problem. This data suggests that an emphasis on user education may be the most effective means of reducing the illegal copying of software.

The Cost of Piracy to the Software Publisher:

Since the early 1980’s, software has been mass-produced for the microcomputer market. Although the licensing and selling of these software packages tends to be relatively inexpensive, quantity sales more than make up for the low unit prices. However, with software’s potential for profit have come increasingly serious problems of piracy, theft, and counterfeiting of software. In the international markets, piracy is even encouraged by some governments (Hubbard, 1991).
Some researchers have claimed that there may be advantages in allowing piracy to continue. It may be that many of those who purchase software would not have been prepared to pay the full price had they not anticipated making copies for other machines (Maude and Maude, 1984). LaRue (1985) argues that software publishers would be better off if they adopted the "shareware" marketing strategy of encouraging users to make copies and distribute them to friends. The idea is that if additional users try it, they might like it enough to purchase their own copy complete with documentation and publisher support. This is not a widely-accepted marketing strategy, however.

Piracy is exactly the same as stealing property (Pressman, 1985). Sale or distribution of a copy of proprietary software is as illegal as the copying of recorded music or the copying of copyrighted books. However, like these other forms of copy violation, software copying has been nearly impossible to stop (Glass, 1985).

Software piracy has lead to huge losses for software publishers, but they seem to be unable to find a solution to the problem. By 1985, publishers were losing $800 million a year to software piracy (See Table 1). According to the Software Publishers Association (SPA), a conservative estimate of $2.4 billion in domestic sales, and $7 billion in worldwide sales of business software was lost to piracy in 1990 (COMDEX, 1991). Piracy costs are estimated to be about 60% of domestic software sales, which is more than $4 billion a year. To date, publishers in the United States have lost over $14 billion in revenues to pirates (COMDEX, 1991). Piracy has a debilitating impact on publishers, particularly the smaller ones, causing some to go out of business (Grover, 1989). One United States software reseller stated that he was able to identify $1 million in business lost to pirated software sales for a one-month period.
The Cost of Piracy to the System Manager

Clearly, software publishers have been losing a great deal of money in sales due to piracy. Software users and managers of users also must be aware of the problem, however, because of the significant legal liabilities that may result from piracy. Some IS managers who have failed to secure computer-based information have been held both accountable by the organization and legally liable (Bequai, 1984; Brinkman, 1983; McKibben, 1983).

In 1989, for example, the Business Software Alliance (a branch of the SPA) successfully sued a large corporate user in Italy for software piracy where roughly 50 illegal copies of Lotus’ “1-2-3” spreadsheet and 25 copies of Ashton-Tate Corporation’s database software were made. Only one copy of each had been legitimately purchased (Hubbard, 1991). In February 1990, the SPA brought a lawsuit against the University of Oregon for copyright violations concerning software from Lotus Development Corporation, Microsoft Corporation, Ashton-Tate, Aldus, Claris, and Wordperfect Corporation. The suit alleged that the University’s Continuation Center (the largest computer training center in the state, serving about 300 to 500 students a month) employees made unauthorized copies of the publishers’ software and training manuals. The university agreed to pay a fine of $130,000 and organize and host a national conference of copyright law and software use. This was the first suit to be brought against a higher-education institution. In May of 1991, Parametrix settled a suit by paying a $350,000 fine plus attorneys fees to SPA. The executive director of SPA commented (Feldman, 1991):

The circumstances that led to our lawsuit against Parametrix are very common. Parametrix is an honest and upstanding company that simply lost sight of its personal computer management. As companies grow, they need to educate their employees and conduct periodic audits of their PCs.

The number of computer-related lawsuits increased from about 600 in 1981 to 3,500 in mid-1984 (Straub and Collins, 1990). The SPA files an average of two lawsuits a week against violators, and their audits for publishers over the last three years have led to fines totaling between $2 million and $3 million (the average settlement over the three years has been between $100,00 and $200,000) (Doyle and Clancy, 1991).

In conducting an audit, the SPA seeks federal permission to stage a raid on a potential violator (disgruntled employees or consultants turn in most violators), and accompanied with marshals, searches the company’s disk drives for pirated copies of the publishers’ software. Only one raid (out of seventy) has turned up no pirated software. Thus, managers should be concerned about software piracy not only because of the potential for substantial monetary fines but also because of the disruption to the organization’s operations such audits may cause. Pirates face fines equal to the list price of the software, destruction of the pirated software, and payment of attorneys fees. In addition, they have to purchase the pirated software from the publishers (O’Conner, 1991). Future penalties may be even more severe.

Some publishers are more aggressive than the SPA. Since 1988, Autodesk (which makes computer-aided design software for engineers, architects and drafters which retails for about $3,500 per copy) has recovered $5 million in lost product sales at a cost of $1.5 million, and still has about 280 active investigations. Autodesk claims that seven to ten copies of its AutoCAD program are illegal for every one purchased.

Because of these fines, and the potential for fines, some large users are doing their own audits. In this way, they can find any pirated copies on their systems and “clean up” their systems at a much reduced expense. Some organizations have built a substantial business around doing audits for customers (Doyle and Clancy, 1991). Others have developed tools for resellers and developers to use when auditing clients. One such tool helps to build software metering systems to monitor the use and distribution of software.

Methods of Protection Against Piracy

The best solution to software piracy would be one that is completely invisible to the user, and prevents piracy 100% of the time. Yet, no such method is available. Instead, there are seven major categories of
protection methods that are used by software publishers (legal protection, copy protection, hardware keys, software keys, embedding of user data, registration, and user integrity). Other categories, such as cryptographic techniques, are generally part of one of the categories discussed (e.g., hardware keys and software keys). The categories are not mutually exclusive, and therefore some software publishers may use multiple methods.

I. Legal Protection

Perhaps the only effective protection scheme is the threat of legal enforcement. “The only viable solution to software piracy is peer pressure, reinforced by lawsuits against major offenders” (Fawcette, 1985). This category is also the most important to managers because all forms of legal protection now make management liable, even when they have no knowledge of the piracy. The liability of management applies if they had the right and ability to control the activity that results in piracy and had some financial interest in the act of piracy (Copyright Law, 1984).

Government policies may be the main protection for publishers against piracy, especially in the international market (Parks and Fleck, 1988). Although this may be true in the international market because of political concerns, domestically, government policies often may be too little too late. Many software publishers may be reluctant to rely solely on legal protection because the publisher must find the pirates and prosecute them at a lower cost than the recoverable fines (unless the publisher wants to make an “example” out of a few pirates to scare the remainder). The cost of litigation against many small infringing pirates may make it impractical to try to find or prosecute them. So, publishers may resort to going after only larger corporate offenders who are worth the cost it takes to enforce the laws. In addition, since fines are directly proportional to the list price of the software (with the exception of those provided under the Racketeer Influenced and Corrupt Organizations Act of 1970 — RICO), if the software sells for a relatively small price (e.g., $50), it is not likely that the publisher will find many instances of piracy where the benefits exceed the costs.

The legal protection category covers four major governmental areas (copyrights, patents, RICO, and trade secret laws) as well two other areas (publisher’s licenses and legal contracts).


2. Patents: Until the 1980’s, patents were not available for computer software. In 1981, the Supreme Court ruled that computer programs are patentable (Straub and Collins, 1990). Because the patenting of software provides the highest legal protection available (Straub and Collins, 1990), it is increasing (Jakes and Yoches, 1989). Patents provide strong protection because they can be infringed upon by the independent development of equivalent programs, even if these development efforts do not produce exact copies (Straub and Collins, 1990).

However, patents require two to three years to obtain, and then last only a few years (3 1/2 to 15 years in the United States). Other drawbacks to securing patents include the considerable expense, and (since they require full disclosure of the software) loss of trade secret protection (Parks and Fleck, 1988; Jakes and Yoches, 1989).

3. Trade Secret Laws: Trade secret laws are state laws that do not require disclosure (as with a patent) or registration (as with copyrights), but both state and federal courts enforce these laws (Lautsch, 1985). Even widely-distributed computer programs do not forfeit trade secret status when the license agreement places restrictions on use and disclosure by the user. Trade secret laws provide additional protection because they can protect ideas, information, and innovations. Since both state and federal courts enforce these laws, publishers are usually near the proper court, if legal action is required. However, trade secret laws have very limited benefit for publishers who sell overseas. International law does not offer the same protection as the United States’ law (Parks and Fleck, 1988).

4. RICO: On occasion, organizations have resorted to using the Racketeer Influenced and Corrupt Organizations Act of 1970 (RICO) (Gemignani, 1988). The intent of RICO was to combat organized crime, and it was a part of the Organized Crime Control Act of 1970. Some publishers claim that the (c) and (d) sections of RICO apply to software piracy. RICO contains a provision which allows any person with injuries resulting from a violation of these sections to sue in federal court and to recover treble damages and reasonable attorney’s fees. Other legal recourse (e.g., copyright laws) provides only for the recovery of provable damages caused by the
breach of contract, and each side must pay its own attorney’s fees.

5. Publisher’s License: Other legal protection comes from the publisher’s license that users sign. Virtually every publisher has some form of policy against unauthorized copying, which is usually stated on the license. A signed license becomes a legal agreement between the user and publisher, where the publisher holds the user legally responsible for any pirated copies that the user makes, or allows to be made. The publisher’s license provides the opportunity for educating the user of the illegality of piracy and the consequences of violating that particular clause of the license. A license is reasonably inexpensive and easy to get into the users hands. Although the publisher’s license does provide some legal protection, pirates are almost impossible to identify. If the software is copyrighted, the protection afforded by a license is nominal, and the license can only serve as an education tool.

More publishers are moving to concurrent site licensing, since that method gives users the most freedom to copy and distribute software (Wasch, 1991). A concurrent site license is a single license with a large user that allows them to use a maximum number of copies (e.g. fifty copies) of the software. The license allows the user to have that many different users use the software concurrently, and there can be different users or computers each day as long as the total number of users doesn’t exceed the number licensed. Users can then treat software as they do any shared peripheral and buy only the copies they need at any one time. Concurrent licensing does conflict with the publishers desire to sell a copy to every potential user on a network, or to each desktop computer. But both interests seem to be best served by a system of concurrent licensing. This seems to be an equitable solution to the problem of piracy (for the publisher) and the need for portable copies of the software (for the user). This recommendation mirrors some of the current policies of the larger publishers. Microsoft adopted a system of concurrent licensing in May, and at COMDEX in November, Wordperfect announced a similar policy. Lotus, Symantec Corporation and Software Publishing Corporation are also beginning to use concurrent licenses. Although the details of each policy may vary, the industry appears to be building a consensus around concurrent licensing as a standard means of licensing applications software on a network. However, concurrent licensing cannot be supported outside of a network, because without a network, there can be no verification or lockout of unauthorized users. Concurrent site licensing is usually incorporated with one of the hardware key methods (see below), or with a software metering system in order to limit the number of concurrent users.

6. Legal Contracts: Some publishers supplement the laws and license agreements with a legal contract. This has been used by large organizations (e.g. IBM) when entering into cooperative efforts with other large organizations.

II. Copy Protection

Software publishers have used copy protection for many years. The most common form of copy protection allows the user to only copy the diskettes (software) once. Recently, a new type of copy protection called a “worm” has been used to protect software. A worm “eats” the target disk when a user tries to copy the original disk. This can be disastrous if a pirate tries to copy a disk illegally to their hard drive. However, the software publisher risks a lawsuit from the user when such disastrous consequences occur.

Because most of the copy protection schemes are designed so that users cannot make a backup of the software (Grover, 1989), users may not have anything to fall back upon if something happens to the single disk. Protected software is more fragile than unprotected, and is more likely to be affected by system changes (e.g. faster CPUs). For disk crashes or computer upgrades, it could be that the software will not be transferable to other media (Grover, 1989). Copy protected software is more expensive to produce and maintain, and legitimate users get this cost passed on to them instead of the pirates.

A few businesses have designed software with the express, open purpose of “unlocking” the various copy protection schemes on the market. Examples of this are “COPY II PC” and “COPY II Plus” from Central Point Software. Another example was Omega Microware’s “Locksmith”®. This may send a signal to users that piracy is an acceptable practice, even though these unlocking programs carry disclaimers and warnings about the illegality of copying software. Both Lotus and Ashton-Tate have abandoned strict copy protection schemes in favor of user integrity. Lotus went so far as to issue lock-breaking software to current owners of “Lotus 1-2-3” versions 2.0 and 2.1. The reasoning was that no copy protection scheme was unbreakable, and no copy protection scheme would discourage determined commercial software pirates (Pallato, 1988). The SPA has said for years that there is no copy protection scheme that cannot be broken. This shift has created some confusion in the marketplace by suggesting to users that it is alright to pirate software, according to a spokesman...
for Lotus (Palatto, 1988). In conclusion, the copy protection scheme doesn’t really prevent piracy, since someone will break the scheme (and confuse the issue of the illegality of piracy as perceived by users), and it may impose an unacceptable burden on the legitimate user (White and Comerford, 1990).

III. Hardware Keys

As early as 1981, publishers introduced the idea of hardware keys (Morgan, 1981), and they soon became a popular method of protection. A hardware key is a connecter device (similar to the connecter section on the end of a printer cable) that attaches to the parallel port on the back of the microcomputer. The software works with the device to limit the number of users who can use the software to the number licensed. Without the device attached, the software will not work. This method does allow the lending of a software package and hardware key to someone else. But as the package can only run on one computer at a time, this level of security not only is acceptable, but may be ideal in solving some of the piracy problems. One supplier claims to have 6,000 software publishers worldwide using their hardware keys. “Nexpert Object” (an expert system software program) is one example of software using hardware keys.

Hardware keys have several advantages. If more than one hardware key is necessary, they plug into one another, so that multiple keys can use the same port. The publisher does have to program the software and the hardware keys to prevent unauthorized use of the software, but any software can use hardware keys. Hardware keys prevent unauthorized use of software effectively and reliably, particularly for networks.

Hardware keys also have some disadvantages. Hardware keys are most suitable for large network systems. Some of the earlier problems with multiple keys and printer conflicts (since it usually uses the parallel port) have since been overcome. For example, keys are now available for serial or parallel ports. Even with this, a hardware key still ties up part of the computer’s hardware, and can require a hardware (port) expansion before being able to use the key. Also, there is a possibility of memory contention with hardware key programs and RAM. In general, hardware keys are more expensive than the other categories and more inconvenient for the user.

IV. Software Keys

Once software registration is complete, the publisher may send the user a “software key” to unlock the software. One common software key is a unique password. Until the password is sent to the user, the software limits the number of times the user can access it. Thus a pirated copy of the software has a very short life. Another type of software key, called an action trigger, uses some method to validate the authorized use of the software. Usually the software checks the date (from DOS and RAM) and determines if the user has authorization to use the software. If it hasn’t been registered and the trigger is deactivated, at some future date, the trigger produces some catastrophic result (e.g., software lockup, erasure of software, etc.). “PC SAS” uses the action trigger type and “Check it LAN” from Touchstone uses the password type.

Software keys provide a deterrent to unauthorized use. They are better than hardware keys in that no additional hardware is necessary for the user to implement the software, which reduces the hassle for the user. It is even less expensive than hardware keys and is therefore an attractive method for protecting software. However, when software keys use passwords, they have to be memorized or written down. Writing down a password, particularly if it is easily accessible, causes a loss in the security of the software. Also, it can only deter unauthorized use of the software, and does not deter piracy. In fact, the user can copy both the software and the password for a pirate. Since it relies on the user to some degree, the publisher is back to depending on the integrity of users.

V. Embedding User Data

Another way to protect software is to require the users to forward their names, addresses, etc. with the purchase. This is “embedded” in the software code and used on all the printouts and screens. Thus, a pirated copy will show the original user’s name and address on all the pirate’s printouts and screens. Drake’s tax software, Cobb’s dental software, and Southern Automated Systems Incorporated’s “T*H*E* RTO” program all use the embedding method of protection.

A display of the name of the purchaser on the screen or printouts creates a climate of awareness, and in particular may make the purchasers less willing to pass a pirated copy on to a friend if their names will be constantly displayed on any further copies made. Embedding costs the publisher very little, and does offer some protection. In particular, it provides a reasonable amount of protection in vertical markets (a market for a specific industry or type of business), versus software with a broad appeal. All the software examples given above for this method are vertical markets.
However, embedding the user’s information provides limited protection, and does not prevent pirates from making copies. It’s primary usefulness is in vertical markets and small user organizations. In particular, embedding can be ineffective if the organization has branches but only buys one copy of the software.

VI. Registration

Many publishers count on user registration. If a user who calls for support does not have his software registered, then the publisher suspects piracy. This gives the publisher some leverage over the suspected pirate, and may even help the publisher to obtain a name and address. At least pirates cannot receive support from the publisher. Most publishers try to register all of their software and use that as a means to validate users calling for support or requesting upgrades. For example, Great Plains Software uses registration as their primary method of protection.

Like licenses, registration is very inexpensive, and it provides publishers several advantages at once. It enables them to make sure that resellers are being honest in their sales, and therefore, the publisher will get their royalties. It also gives publishers a list of all their customers (if a reseller goes out of business) for use in direct upgrades. Publishers can track the number of registrations returned against the number shipped to validate missing software packages. Since the publisher wants several of these advantages anyway, registration will continue to be an important part of selling software regardless of any other protection method used in the future.

Registration helps to deter pirates from getting support, but does little else. Pirates can make all the copies they want of off-the-shelf software, especially those software packages that are user-friendly (where support and manuals are less necessary). Another problem is getting users (or resellers for users) to return the registration to the publisher. A spokesman from WordPerfect Corporation claims that only a small fraction of legitimate users ever send in their registration cards.

VII. User Integrity

Perhaps the most common scheme is to rely on the integrity of the user. In other words, the publisher uses no scheme, but simply relies on the awareness of the user (that piracy is illegal, and what constitutes piracy), and the integrity of the user to not allow any unlawful copies of the software to be made. For example, Central Point and ChipSoft rely largely upon user integrity to prevent the illegal copying of their software.

Obviously, user integrity costs the publisher nothing in the software package; rather, the costs are opportunity costs created by losses due to piracy. Because it is so easy for the user and so simple for the publisher, many publishers depend upon user integrity. However, user integrity probably offers the severest limitations. The burden of adhering to these policies is on the user. This solution gives users the opportunity to continuously add to their library of “free” software by using the “copy” command.

Survey Results

In our survey of 13 software publishers, one question (#4 in Appendix A) asked for the method of protection that the publisher uses now (see Table 3 and Figure 1). The legal category is available to every publisher,
and 11 of the 13 use a license with a clause about the illegality of piracy. Eight used nothing (user integrity), 4 used embedding, two used registration, and one used a software key. Some software publishers used multiple protection methods. This small survey shows that most publishers are probably relying on the most inexpensive protection schemes (licensing and relying on the integrity of the user). Question #5 of Appendix A asks which protection method is the best. Only 9 of the 13 publishers responded, and the answers varied considerably (see Table 3). This suggests that publishers do not recognize any one method as being best. Also, few of the publishers used the protection method that they identified as being “best.”

In response to the questions on ways to stop piracy (#6-9), one commonality was an interest in educating users. These comments ranged from advertisements in magazines to building awareness of the problem. Regarding what could be done to better protect publishers (#6), three publishers answered legal action would better protect them, and only one felt that nothing could be done. Most responses relate to technological changes (e.g. hardware changes, operating system changes). On the sufficiency of laws regarding piracy (#7), six believed that the laws are sufficient, four said insufficient, and three felt laws were useless. The responses to the question about what could be done to catch pirates (#9) included SPA audits, a reward system for “whistle blowers,” and education.3

On question #14 (which group contributes the most to piracy?), respondents stated that 64% of software piracy is attributable to small users, 25% is attributable to medium users, and only 11% is attributable to large users.

Another solution to software piracy publishers are using is to see what they can do to get resellers more involved in the process of limiting piracy. One way is to use a form of soft dollars or marketing funds as an incentive for resellers to not only educate customers but to more closely track software sales, including upgrades. The reseller becomes an active watchdog for publishers, knowing that they will be financially rewarded for their efforts.

**Conclusion**

The combination of large volume markets, the ease with which programs can be reproduced and distributed, and the small risk of being found out or sued has led to the rise of a black market of unknown size in pirated copies of popular microcomputer software products.

Straub (1990) and Herron and Witt (1991) suggest that a user and IS manager education program is necessary to alleviate the piracy problem. Our survey confirms that many software publishers believe the same thing. Executives may not realize employees have been copying software. The employees, in turn, may not realize that making a copy of the software for themselves or a friend violates the copyright law. This environment makes education essential for the successful discouragement of piracy. Also, education may be a cost-effective means of reducing the problem. Any of the technical protection scheme categories tend to add about 25% in development time for programming (Grover, 1989). An
effective education plan, especially if publishers only depend on user integrity, could mean a significant monetary savings in software development.

For the large and medium users, concurrent site licensing might help, although it offers virtually no help for the small user. The continuation of audits by independent publishers and associations (such as the SPA) should contribute to a decline in piracy, but again this principally applies to larger users. A useful education program should contribute to a decline in piracy for these two groups as well. In particular, IS managers have the opportunity to increase awareness and education for users in the medium and large user environments.

If the survey is correct, 64% of all piracy is done by small users, and neither of the above solutions would affect most piracy. Until publishers come up with a more effective method of preventing piracy by the small users, an education plan seems to be the most important part of any solution to preventing piracy in the smaller user group.

**APPENDIX A**

1. What will your software sales be in 1991? $ ________________

2. What is your estimate of lost revenues this year due to piracy? $ ______________

3. Since 1985, is piracy ...
   [ ] increasing each year. [ ] about the same as before.
   [ ] decreasing each year. [ ] don’t know

4. What is your software protection scheme?
   [ ] Embed company information in the software before delivery to user.
   [ ] Allow only one installation of the software diskette.
   [ ] Use password. How? ________________________________
   [ ] Hardware keys for software.
   [ ] Depend on the integrity of the user (nothing).
   [ ] Other: ______________________________________________

5. What is the best protection scheme?
   [ ] Ours [ ] ____________________________________________

6. What should be done to better protect you from piracy?
   [ ] Legal action [ ] Hardware changes
   [ ] Operating systems changes [ ] Nothing
   [ ] Distribution channel changes [ ] ______________________

7. Are laws regarding piracy...
   [ ] sufficient? [ ] too stringent?
   [ ] insufficient? [ ] useless?

8. What could be done to better enforce the laws against piracy? ________________________________

9. What could be done to help catch pirates? ________________________________

10. How could user audits be done efficiently and effectively? ________________________________

11. Does your software license have a clause concerning piracy?
    [ ] No
    [ ] Yes .. What? __________________________________________
    What is the punishment? ________________________________

12. What is your policy for multiple users within an organization?
    [ ] Sell licenses one at a time. Discounted? [ ] Yes  [ ] No
    [ ] Sell multiple user licenses in increments. Of _____.
    [ ] Use site license for maximum number of users.
    [ ] Other ____________________________________________
13. Do you have a network version? [ ] Yes  [ ] No  [ ] No, but will have ____________

14. What percentage of piracy would you attribute to each group?
   ___%  Large Users (Fortune 1000 etc.)
   ___%  Medium Sized users (Businesses, Government Agencies)
   ___%  Small Users (Personal use, Home use, Small Businesses)

Endnotes
1The U.S. Senate Judiciary committee approved a bill that would impose criminal penalties for software copyright infringement, including a fine of up to $250,000 and a prison term of up to 2 years for illegal reproduction or distribution of 11 to 49 software copies.
2 Omega Software has since gone out of business trying to keep up with the many protection schemes.
3For example, the SPA sponsored a billboard at COMDEX 1991 in Atlanta which read “Copyright software illegally and you’ll get this hardware absolutely free’” showing a pair of handcuffs.

References

Comdex Show Daily, (May 26, 1991) p.3
Fawcette, J.E. Fighting common sense. INFO WORLD, (March 4, 1985), 5.
Feldman, S. “Legal beat.” PC Sources (September 1991), p.64


Tom Singleton is currently a graduate student in the School of Accountancy at the University of Mississippi.

Milam Aiken is an Assistant Professor of Management Information Systems in the Department of Management and Marketing at University of Mississippi.

Ashraf Shirani is currently a graduate student in the Department of Management and Marketing at the University of Mississippi.
Related Content

A Theoretical Explanation of the Evolving Role of Users in Shaping Corporate Information Systems
www.irma-international.org/article/theoretical-explanation-evolving-role-users/55690/

An End-User’s Journey of System Use: A Change In Attitudes And Behavior Over a Period
www.irma-international.org/chapter/end-user-journey-system-use/53088/

The Role of Trainer Behavior in End User Software Training
www.irma-international.org/article/role-trainer-behavior-end-user/3746/

The Impact of Local Area Networks on Users and Their Work: A Field Study
www.irma-international.org/article/impact-local-area-networks-users/55660/

Errors in Operational Spreadsheets
www.irma-international.org/chapter/errors-operational-spreadsheets/53093/