Dear Editors,

Jean Camp’s “Access Denied” article (IEEE Security & Privacy, September/October 2003, pp. 82–85) set out to list some dangers that digital rights management (DRM) has for creators. As a consumer and producer of intellectual works, I eagerly read the article. However, I was left nonplussed.

The article contains many blunt assertions of DRM system failures followed by unsupported examples instead of explanations. For example, Camp claims that “‘owned’ code can’t be subject to journalistic or technical inquiry,” despite the fact that open-source and closed-source software, both of which are frequently owned, are often reviewed and inspected, an act that is protected by subsections (g) and (j) of section 1201 of the Digital Millennium Copyright Act (DMCA).

The biggest fear in the article, and also the easiest to dispel, is the threat of authors losing the right to “publish, review, alter, or even read their own works” if, as an example, DRM-protected formats are used by word-processing software. Ms. Camp has this threat realized through two fronts: other software accessing said formats being illegal, and terminated employees losing use of “their” licenses of the software.

The issue of inaccessibility is a paper tiger, and it is because of the DMCA. Subsection (f) of section 1201 plainly indicates that reverse-engineering for the purpose of interoperability is allowed. The US Copyright Office’s declaration in the Lexmark printer case made this point clear. (Granted, at publication time of Camp’s original article, the Copyright Office had not yet issued its ruling; what was fresh in people’s minds was a US District Court’s determination that Static Control Component’s reverse-engineering defense was insufficient to stop an injunction.) The fact that the DMCA already includes a subsection protecting reverse-engineering makes Ms. Camp’s unreferenced statement that “the US Copyright Office’s most recent review of the DMCA found no reason to provide an exemption for reverse-engineering” all the more puzzling. Did Camp mean that the office had refused to expand the existing protections for reverse-engineering? If so, perhaps the reason is because the office already knew how they were going to respond to the Lexmark case, and found the existing safeguards sufficient.

The second front of the supposed assault on creators’ freedoms was that a fired employee loses access to works created with the company’s software. This is also a perplexing argument, but it might have to do with cultural issues. For starters, employees in a work-for-hire environment typically do not own the copyrights of the works they create. I realize that this is not a popular concept in academic circles, but the private world has dealt with it just fine. Groups of people often create works that are not owned by any individual—in stark contrast to Camp’s image of “a lone creator.” Regarding works produced for the organization, an individual typically is intended to lose access to them after termination of employment, being required to return or destroy all personal copies. The only way you would lose access to individually owned works, such as being blocked from “[reading] your own resume or [obtaining] your list of contacts’” would be if you kept the only copy at the office. Again, someone from an academic background probably is more casual about the typical boundaries between home and work life. Inability to read a data format is the least of your problems if you’re keeping your only copies of vital personal data at the office.

But irrespective of what DRM-protected formats exist, users are free to use other software to create their own new documents. And even if it were illegal to interoperate with the protected formats (which it isn’t), the DRM-enabled word-processing software is guaranteed to have methods for exporting the data in simpler formats, such as publishing as HTML. At the very minimum, the document can be printed out on paper, from where it can be typed back into another piece of software. (Re-entering CAD diagrams from paper would be more tedious, but I hope the example suffices to show that such workarounds are not limited to word-processing software.)

There are legitimate concerns to be made about DRM, and IEEE members need and deserve to be well-versed on the matter. To have an influence on our government, at a minimum we must speak from knowledge of how the current system operates, of what’s actually allowed and prohibited. I look forward to future articles on the subject that examine a narrower score of DRM, but with more accuracy.

—Daniel Weber
djweber@alum.mit.edu

Author Jean Camp responds:
The state of affairs was accurately repre-
Dear Editors,

“Change the game?” (IEEE Security & Privacy, November/December 2003, pp. 62–67) is a fine article. I hope it triggers intense discussion. Here are some of my thoughts.

Security keeps getting worse. I have been working on computer security for 40 years. Despite very real improvements in computer security understanding and practices since that time, the average computer user today faces more threats and more capable attackers, needs to know more about security, and is vulnerable from more directions. Despite defense in depth, formal approaches, correctness proofs, and bug-finding tools, vulnerabilities keep being found.

The “current approach” has had plenty of time to work, and it hasn’t. I was at a security workshop over a year ago that assembled folks to talk about measuring information assurance. The usual suspects attended. The usual approaches were proposed. A distinguished colleague pointed out that everything being said had been said 20 years ago, and asked me to confirm—I said it was at least 30.

I like the notion that behavioral science should be consulted. I wish there were a well-founded theory we could apply that would suggest anything new: Perhaps we could take a belief logic like Lampson, Abadi, et al., and add additional “illogical” rules of inference to simulate social engineering. There is a notion in the quality area called “poka yoke,” or “mistake-proofing,” that suggests, for example, eliminating the ability to set a switch improperly by removing the switch. (See www.swmas.co.uk/Lean_Tools/Pokayoke.php.) The privacy application might be Scott McNealy’s “get over it.”

Can we think of some other way to change the game we’re playing so that we have some chance of winning?

—Tom Van Vleck
thvw@multicians.org

No clear answers

Dear Editors,

I finally had a few minutes to read through the November/December 2003 issue of IEEE Security & Privacy. I was happy to see dueling articles by Dan Geer and Dave Aucsmith (pp. 14 and 15). My personal opinion is that Geer has the upper hand, but that’s not important at the moment. What is important is the following article by James Whitaker.

It is distressing to see an S&P editorial board member take such advantage of his position to disseminate such biased fear, uncertainty, and doubt without an opposing point of view. The title “No Clear Answers on Monoculture Issues” is deceiv-
I still have the original Ada introduction and reference manual from 1980 as well as a textbook from 1983. I also have several late-1970s publications on C (Elements of Programming Style, the Bell System Technical Journal devoted to C, and Software Tools). I purchased them when they were new.

According to these books, the military wanted a common language to replace 400 different—mostly embedded—languages. No mention of C. Moreover, the textbook uses Pascal as the language to show movement to Ada. During this time period, Cobol and Fortran were far more common than C, which had not been established well enough to be a target for replacement. The call from the military for this new, common language was in 1975 as a public competition.

I expect that Whittaker was hoping no one would remember what was going on in the 1970s to ask him to prove what he said given these facts. Especially after he accused that the report’s authors “. . . clouded this important topic with unsubstantiated claims and language that suggested more of an attack against Microsoft . . .” Whittaker has done exactly this in defense of Microsoft.

There is much more to dispute within this article, and someone should be allowed to do it.

—Bob Bruen
bru@coldrain.net

Author James Whittaker responds:
My experience programming with and for the government for more than a decade doesn’t mesh with Bruen’s quotes from government manuals. Every time we started a project, the programmers would lament not being able to use C because of the government’s Ada mandate. Whether the manuals mentioned C or not, the big battle in every project I was a part of was between C and Ada.

But Bruen misses my point: you can give the industry safer programming alternatives (if you don’t accept Ada as such a choice, then how about metalanguage?), but they’ll probably choose the unsafe alternative anyway! ☐

IEEE Computer Society Publications Office
10662 Los Vaqueros Circle, PO Box 3014
Los Alamitos, CA 90720-1314

DEPARTMENT EDITORS
Application Security:
Martin R. Stutz, Air Force Research Laboratory, and James Whittaker, Florida Institute of Technology

Attack Trends/Malware Recon:
Iván Arce, Core Security Technologies, and Elias Levy, Symantec

Bibli Tech:
Marc Dorner, Morgan Stanley

Book Reviews:
Charles Pfeeger, independent consultant, Shari Pfeeger, RAND, and Martin R. Stutz

Conference Reports:
Carl E. Landwehr, University of Maryland

Digital Rights:
Michael Lesk, Internet Archive

Education:
Matt Bishop, University of California, Davis, and Deb Frincke, University of Idaho

Global Perspectives:
Jim Hearn, independent consultant

On the Horizon:
Nancy R. Mead, Carnegie Mellon University, and Gary McGraw, Cigital

Secure Systems:
Sean W. Smith, Dartmouth College

COLUMNS
Privacy Matters:
Michael A. Caloyannides, Mitretek Systems

Clear Text:
Bruce Schneier, Counterpane Internet Security

STAFF
Lead Editor: Kathy Clark-Fisher
kclark-fisher@computer.org

Group Managing Editor: Gene Smarte

Staff Editors: Scott L. Andresen, Jenny Ferrero, and Steve Woods

Production Editor: Monette Velasco

Magazine Assistant: Hazel Kosky

security@computer.org

Contributing Editors: Cheryl Baltes, Keri Schreiner, and Joan Taylor

Original Design: Kraus & Assoc. and Larry Bauer

Original Illustrations: Robert Stack

Graphic Design: Alex Torres

Publisher: Angela Burgess

Assistant Publisher: Dick Price

Membership & Circulation Marketing Manager: Georgann Carter

Business Development Manager: Sandra Brown

Assistant Advertising Coordinator: Marian Anderson

SUBMISSIONS: We welcome submissions about security and privacy topics. For detailed instructions and in formatting, see the author guidelines at www.computer.org/security/author.htm or log onto S&P’s author center at Manuscript Central (www.computer.org/mc/security/author.htm).

For editorial guidelines, visit www.computer.org/security/.

EDITORIAL: Unless otherwise stated, bylined articles as well as products and services reflect the author’s or firm’s opinion; inclusion does not necessarily constitute endorsement by the IEEE Computer Society or the IEEE.