

Full Disclosure¹

by **M. E. Kabay, PhD, CISSP-ISSMP**

**Assoc Prof Information Assurance
School of Business & Management
Norwich University**

How should we handle known vulnerabilities and dangerous viruses? Should we publish full details, conceal some details, or suppress publication that would allow exploitation until patches or updates are available from manufacturers?

At the EICAR conference in Brussels the first week of March 2000, one of the topics discussed by researcher Sarah Ford, PhD (at that time a member of the T.J. Watson Center of IBM) was the range of different attitudes towards disclosure of vulnerabilities and exploits.

Dr Gordon told us that the anti-virus (AV) world and the information security (INFOSEC) worlds differ significantly in their disclosure models.

In general, the AV world frowns on open disclosure of detailed virus codes; in contrast, the general INFOSEC world has developed respected venues for full disclosure of system or network vulnerabilities and exploits (e.g., BugTraq).

Support for full disclosure of such details (down to the source-code or script level) from professional, honest security experts (the Good Guys) is based on subsets of several key beliefs:

- The Bad Guys know about the vulnerabilities anyway
- If they don't know about it already, they will soon with or without the posted details
- Knowing the details helps the Good Guys more than the Bad Guys
- Effective security cannot be based on obscurity
- Making vulnerabilities public is an important tool in forcing vendors to improve their products.

There is indeed a criminal underground where viruses and exploits are widely exchanged and explored. VX (virus exchange) bulletin boards are widespread; in the criminal hacker underground, there are open discussions on IRC (Internet Relay Chat) channels of new methods for breaking into sites and restricted Web sites for exchange of exploits. Since the criminals are really aware of these techniques before system administrators and security experts learn of them, doesn't it make sense -- so the argument goes -- to spread the knowledge where it can do some good? In addition, we know that cryptographic techniques ought to be exposed to scrutiny by experts to avoid failures; why shouldn't other aspects of security be made public too?

¹ This article was originally published in the "Logoff Column" of the *Information Security Magazine* for May 2000. This version has been updated with minor corrections.

Full Disclosure

As for putting pressure on manufacturers, one of my colleagues told me about an incident that illustrates the frustrations that sometimes underlie full disclosure. He informed an important software supplier of a major security vulnerability. The product manager ignored him for a month. At that point, patience gone, my friend informed the product manager that he had exactly one more day in which to produce a patch; otherwise, he said, he would publish the hole in full in the appropriate USENET group. My friend received a patch within one hour.

So why would anyone object to full disclosure of detailed viral code and exploits? The arguments are that

- Nobody except researchers needs to know the details of viruses or even of specific exploits;
- Publishing in full gives credibility to ill-intentioned Bad Guys who do the same;
- Full disclosure makes kids more susceptible to criminals' propaganda about the acceptability of illegal computer abuse.

How, exactly, does publishing the details of a new virus help system administrators? In this view, surely such details should be exchanged only among colleagues who have developed trust in each others' integrity and who have the technical competence to provide fixes. The *zoo* kept by ICSA Labs serves this function: all of the members of the Anti-Virus Product Developers' Consortium have complete access to all the viruses that are contributed; the members sign a Code of Ethics that forbids distributing viruses casually to anyone who wants samples.

Sarah Gordon told us that opponents of this stance see the attitude as arrogant and elitist.

On the other hand, should we publish exploits where eight-year-old children can use them for automated attacks on Web sites? Don't we give naive people the impression that it's OK to publish any attack code, regardless of consequences? What's the difference, then, between publishing vulnerabilities or exploits and actually creating attack tools? Was creating and publishing BackOrifice a morally neutral or even useful act? BackOrifice is a tool that is explicitly designed to install itself surreptitiously on systems and then hide in memory using stealth techniques modeled on what some viruses use. This is a contribution to security?

As for me, I fear that, faced with vendor resistance to fixing security vulnerabilities, we will continue to see users and experts turning to full disclosure of security information even though many of them detest using such tactics.

