

# When Worlds Collide: Information Sharing for the Security and Anti-virus Communities

Sarah Gordon, IBM Thomas J. Watson Research Center <sgordon@watson.ibm.com>

Dr. Richard Ford, Verio Inc. <rford@format.com>

## ***Abstract***

Current trends towards anti-malware software, designed to provide protection from network-aware Trojans, viruses and various forms of malicious active content are bringing the mainstream anti-virus world closer to the more general information security world. At the same time, as information security researchers and professionals begin to investigate the various types of threats posed by active content, we are observing a significant increase in the overlap in areas of influence and interest. While this cross-pollination provides an exciting new source for ideas and innovation, it also poses some novel challenges in terms of differences in mindsets and skill sets. For various reasons, researchers may not be aware of some of these differences. One of the most critical differences, and one that must be rationalized for a successful integration of the two worlds, concerns Information Sharing. In this paper, issues related to diametrically opposed positions regarding information sharing are examined; the reasons why each of these positions has evolved are discussed. Dangers of ignoring the current conflicts are considered, and proposed research that would facilitate the assimilation of the two current paradigms possible is provided. As the worlds of “anti-virus” and “computer security” collide, finding a way for the two groups to work together effectively is paramount if both are to work together toward the common goal of protecting the user.

Keywords: full disclosure, computer security, computer virus, information sharing, group dynamics

## The Ever-merging Worlds

To an outsider, it may come as a surprise that the traditional computer security discipline and the mainstream anti-virus world are not one and the same thing - after all, in many ways, computer viruses and malicious code are just a subset of the more global system security problem. However, currently there is a distinct gap between these two worldviews in several important ways: skill-sets, philosophy, population... the list is long.

In many ways, this somewhat artificial distinction has grown out of the roots of the worlds. Generally, when one considers computer security, one of the primary concerns is the security of network-based machines - particularly servers. Furthermore, the prevalent desktop operating system when the virus problem mushroomed were *MS-DOS* operating systems which have no obvious “levels” of security; users were essentially masters of their machine, and there was no differentiation between “administrator” and “user”. The explosion of the desktop machine has, in essence, split out the “*Microsoft*” world from the security world.

While the rest of security world began to tackle the problems posed by higher connectivity, the real-world security problems within *DOS* were primarily viral in origin. Meanwhile, the virus problem was effectively non-existent in the networked Unix world. Thus, security practitioners focused on ways in which attackers could penetrate systems remotely, or gain unauthorized access levels to local accounts while the antivirus research community developed research methodologies and solutions for the primarily *DOS*-based virus problem. The distinction between the realm of the anti-virus expert and the security expert was relatively clear, and there was little real-world overlap between the groups.

Two circumstances set the stage for the drawing together of these groups - an event that is becoming more evident with every passing day. First, the increase in *desktop connectivity* made the corporate LAN vulnerable to compromises brought about by individual users at a level that had never been experienced before. Second, the “secure” operating system *Windows NT* was introduced.

The addition of security to the desktop suddenly made computer viruses an interesting topic for the security expert. After all, the idea of viruses as a delivery mechanism for new and interesting payloads to the corporate LAN was intriguing. Additionally, as computer viruses become more network-

aware, the desktop security issue can have a massive impact on corporate security: Melissa purportedly resulted in e-mail being shutdown at several large companies [Sullivan, 1999; Defoe, 1999], a direct attack upon availability. The *ADMWorm* had the potential to race across susceptible machines, leaping from server to server [Gordon, 1998]; Caligula used File Transfer Protocol (FTP) to send a victim's sensitive data to a virus exchange site [Patrizio, 1999]. Viruses themselves could be used to decrease a company's isolation from the rest of the Internet, or facilitate compromise of entire networks. Suddenly, viruses were more than a curio in the general security world: they were a threat to general connectivity and network security.

From the viewpoint of the anti-virus researcher, *Windows NT* made a whole wealth of security information instantly relevant. Whereas a virus simply had to malloc a block of memory, copy itself and twiddle a couple of bytes in the Interrupt Vector Table in order to become resident under *DOS*, becoming a permanent part of the *Windows NT* environment meant some work to navigate normal security restrictions; a successful binary file virus for *Windows NT* would have to work out how to jump from a user-level process to another security level. Internet connectivity also brought a need for familiarity with various transport protocols and system security issues into the mainstream antivirus world. Once, viruses could only travel between computers on sneakernet; now increased connectivity brought a raft of network-related options with it.

With the rapidly increasing reliance on both internal and external connectivity, the potential impact of a virus' payload is growing rapidly. Until recently, the most damage a virus could do to a company would be pre-programmed manipulation or corruption of data; now, it is conceivable for an appropriately contrived payload to circumvent the corporate firewall, allowing a savvy intruder open access to the very heart of the company. The availability of "tools" like *Back Orifice 2000*, which allow for the remote control of desktop machines is merely a precursor to what could be accomplished by a virus. Clearly, such a beast would be highly relevant to both the security world and the anti-virus world, as it would both replicate and provide for network compromise.

While the merging of the security world with the virus world fits well into the search for an integrated solution to desktop security issues, there are some key problems to be overcome in order to make this transition. Perhaps the most important is that there are radical differences in the "information

sharing” models employed by some security experts and the anti-virus community.

## **Information Sharing Today**

The Internet has had a vast impact on the ways in which information can be passed between groups, as well as the accessibility of computer systems by untrusted third-parties. While virus researchers are still coming to grips with what this means in terms of the viral threat (the rapid spread of Melissa may well indicate the shape of things to come in terms of network-aware viruses), the security world has been dealing with these issues for some time. Exploits can be sent to thousands instantly, and there is a ready availability of tools that can be turned against the machines of unsuspecting system administrators.

Due to this rapid dissemination of information, we believe that the future of information sharing in both computer virus and security terms is Internet-based. While there is always a need for reference publications such as *Virus Bulletin* or refereed Conference Proceedings, information regarding the “latest” or current threat is likely to be disseminated on an increasingly real-time basis, if only because the threat is becoming increasingly real-time in nature. For this reason, we shall concentrate our discussion around Internet-based communication, and examine the most important information sources for viruses and more general computer security. Thus, for the purposes of this paper, we shall define Information Sharing to be the disclosure and discussion of security vulnerabilities and technical virus information on the Internet.

Information about viruses can be obtained from a variety of online sources. Based upon our experience, the primary sources are vendor-sponsored WWW Sites, where news about the latest viral threats, virus detection signatures and virus descriptions are made available. In addition to this, the Usenet newsgroup comp.virus is monitored by some system administrators tasked with keeping up to date on virus problems. *CERT* and *CIAC* provide some information about viruses and virus hoaxes; however, the product vendor appears to be the main source of information. The information provided by vendors, comp.virus, *CERT* and *CIAC* have been provided from a “limited disclosure” perspective; that is, viruses are described in general terms, and their source code is not made available. In the mid-1990s the anarchic, unmoderated alt.comp.virus hierarchy began to grow, bringing with it an increased availability of virus code to the general user. At the time, many researchers were aghast, and felt that anyone taking part in this

group or its children was a “black hat”. While many antivirus researchers now take full part in alt.comp.virus, the group is still not wholly accepted by the community.

Analogous to the anti-virus sources listed above are a number of net-based resources for gathering information about security matters. These sources can be broken down into two primary classes (although the dividing line is not always clear): those that deal with the immediate problem, and those that discuss longer-term, wider or more philosophical/theoretical issues. Examples of immediate problem resources are *CERT Advisories*, *Bugtraq* and *NTBugtraq* mailing lists. In each case, the source is centered primarily around the task of alerting users and administrators to *current* issues concerning security - that is, they are lists of *vulnerabilities and exploits*. They also take a longer-term view, providing archives of previously published information

For instant access to information, many administrators turn to *Bugtraq* and *NTBugtraq*. Listed as resources by such well-recognized sites as *COAST* and *CERT*, these lists provide administrators with the very latest information on exploits. Over time, these moderated lists have become the *de facto* standard for discussion of current problems and solutions; indeed, many cite *Bugtraq* as a proving ground for security techniques [Cowan *et al.* 1998] or simply a resource for gleaning source code [Cohen, 1996]. Similarly, *Bugtraq* is cited by [Denning, 1999] as a “must read for Unix System Administrators”, as well as being featured in the “Major Mailing Lists” section of *Practical Unix and Internet Security* [Garfinkel & Spafford, 1999].

In information sharing terms, there are several important differences between *CERT* and *Bugtraq*; here, we list the ones that are most relevant to the topic of discussion. First, *CERT* concentrates on more global issues - that is, issues involving multiple incidents. Additionally, *Bugtraq* and *NTBugtraq* are interactive - that is, there is a ready two-way interactive exchange of information. Finally, there is the issue of disclosure - *Bugtraq* and *NTBugtraq* practice forms of “full disclosure” of information, whereas *CERT* does not.

## **Disclosure Positions and Arguments**

Full Disclosure proponents argue that by providing as much information about a potential problem as possible, system security administrators can make well-informed, intelligent decisions regarding the risk to their particular system, and decide what actions they should take. Additionally,

both *Bugtraq* and *NTBugtraq* moderators believe such publication can force recalcitrant vendors to fix problems which they might otherwise choose to ignore until a more convenient time [Levy, 1999; Cooper, 1999]. Finally, full disclosure is also seen as a way to expose more people to the technological problems, and possibly enable others to come up with solutions or facilitate the sharing of further information [Cooper, 1999].

However, these benefits of full disclosure are not embraced by all in the mainstream security world. Proponents of the “no disclosure” model argue that security information, particularly with respect to vulnerabilities, is best kept within a small trusted circle. They argue that only harm can come from the dissemination of this information: holes will be exploited, machines compromised. The “right” way to approach the problem is for users to keep systems patched to the latest version of program code. The benefit of this approach is that fewer people are in a position to exploit the vulnerability. The bad news is that until a formal fix is released and rolled out, thousands or even millions of machines may be vulnerable. Simple steps to limit the scope of the problem may have been missed simply because of the lack of disclosure. Ultimately, the extent of the practice of “no disclosure” within the security world is, by definition, not known: it is very hard to prove whether or not a vendor had foreknowledge of a particular vulnerability.

Partial disclosure in the security world attempts to solve the problems posed by no disclosure, without giving away the benefits. However, this is not always possible. Consider the following hypothetical warning: “The foo program has been determined to be vulnerable to a buffer overflow. Users are urged to update to version x of the foo program. If an update is not immediately possible, suid permissions should be removed immediately.”

Such a warning does indeed address many of the “no disclosure” problems. Users are warned; remedial action is given when possible – administrators are warned without giving out exact details of the exploit. However, even given somewhat sketchy details, a programmer can probably deduce and replicate the exploit discussed above. This is especially true in the Unix environment, where source code to popular utilities is often made available.

Thus, it is apparent that there is still a lot of debate concerning the “right” way to deal with information related to vulnerabilities. In his Ph.D thesis, [Howard, 1997] states:

*Disclosure of vulnerabilities is more difficult to agree on. If both the existence and the technical details of all vulnerabilities were fully disclosed, this would undoubtedly result in suppliers making a greater*

*effort to secure their products. This would be because more attackers would probably be exploiting these vulnerabilities. As to whether this would lead to more or less security is unclear (and hotly debated).*

Whereas it is not uncommon for “white hats” to publish fully working exploits within the security world, similar behavior within the anti-virus community is strongly condemned by many: most members of the anti-virus community practice partial disclosure at best. The reasoning behind this stance is that while there are several arguments for the support of full disclosure with respect to exploits, publishing virus code is seen as “different”, and as accomplishing little in terms of providing protection to the users.

Part of the reasoning related to differences may stem from the fact that viruses are autonomous – that is, once released they can spread from machine to machine without any harmful intent on the propagators. Thus, damage can result from a virus sample without any direct action by a perpetrator; simple carelessness or inexperience is enough to release a “Zoo” virus into the wild. When one considers a security exploit, the code itself must be “aimed” or directed by an individual. That is, for damage to result, there must be a certain amount of intent on the part of a would-be intruder. Additionally, once that damage has been done, the exploit must again intentionally be aimed at another target: there is no self-replicative property. Therefore one can argue that while a virus made publicly available represents a more immediate and lasting danger, an exploit still requires hostile intent in order to be used.

Another possible cause for this different approach is that the vast majority of computer viruses take advantage of the same system functionality, whereas demonstrating different vulnerabilities within the general security arena generally involves essentially different exploits. Thus, it can be argued that publishing viruses generally merely underlines an already understood (and potentially unsolvable) problem, whereas publishing an exploit may alert the user community to a new, solvable problem. To quote one antivirus researcher:

*Unlike bug exploits, where at least a case may be made that it's a valid last resort if a vendor has been notified of a bug and ignored it, viruses don't go away when you just fix a bug. [Chess, 1999].*

Some researchers would assume the position that viruses do not “exploit” any security vulnerability at all: they work perfectly well within the defined access-control limits of a particular user or process. Thus, there is no

“vulnerability” to be fixed, and hence nothing at all worthwhile in publishing another computer virus.

Finally, in the security world, there are strong advantages in keeping up with the latest exploits. As an administrator, one can usually limit the use of a particular exploit, either by disabling a service or reconfiguring it.

Ultimately, a skilled administrator will use knowledge of the exploit to expose those users who are attempting to penetrate his system by monitoring his network for signs of anyone using the exploit. Thus, knowledge of a particular technique can be a very powerful tool for securing a network.

This situation is very different from the anti-virus world, where there traditionally has been little or nothing a user can do aside from follow standard computer hygiene and keep up to date with the latest “signature file” from their anti-virus software vendor. Unless a virus is radically different from its predecessors, detailed knowledge of the source code of a new virus is of little help in limiting a company’s exposure to it.

We asked the moderators of *Bugtraq* and *NTBugtraq* for their opinions on the publication of viral source code on their mailing lists. They indicated that despite the arguments for limiting disclosure, they would consider publication on a case by case basis:

*There is, IMO, a distinct lack of understanding of how viruses work within the NT Administrator community. As viruses spread to affect NT servers more aggressively, it is important that they learn something about them to better assist them in thwarting their affects, and possibly, preventing them completely.*

*Application servers, such as Exchange, need to address the virus issue directly. Their inability to withstand a virus such as Melissa or Zipped\_Files.exe speaks directly to the issues that NTBugtraq attempts to resolve.*

*Like exploit code, providing full disclosure on viruses which can affect NT systems would seem to be logical for NTBugtraq. I do not want, however, email messages from NTBugtraq to actually invoke a virus themselves.*

*I will continue to treat the issue on a case by case basis. I have, for example, sent messages to the list simply warning them of some new virus. I have contacted many of the AV vendors and received name and email addresses where I can submit new viruses as I receive them. I usually wait for confirmation from more than one of these*

*before I do anything publicly. And then, if the virus is not specific to NT, I usually will not say anything about it.*

*As far as posting virus code in the future...hmm...I guess it will depend on the virus. I honestly don't feel that Microsoft is doing enough to prevent them, and at the same time, I'm not at all happy with the AV Industry and the way that works. If I do post code in the future, it may be based on a feeling that someone out there has a better way to handle these things, but, today they don't know enough about viruses to say what that better way is. My take on the AV Industry is that it's a closed self-perpetuating group that sees no end to its existence. I've always been working to do myself out of a job, so when I see something that believes itself to be permanently entrenched, I suppose my hackles get up on end.*

*Something has to change, and thus far, it appears that publication in responsible fashion is an effective means of causing that change. What "responsible" means, however, is still not clear. [Cooper, 1999].*

The moderator of *Bugtraq* expressed similar sentiments:

*The only reason multiple viruses got posts to the list is because of Nick's rather smug comments. The idea that because a virus is written different, yet exploits the same vulnerability, it can't be stopped is ridiculous.*

...

*Please note I am not a "full disclosure" extremist. Some people may have this impression because I run the list. I simply use "full disclosure" as a weapon, and in this case the AV community needed to be hit hard on the head with it.*

...

*I see you telling me that viruses and exploits are different because with exploits, they come out, the vendor releases a patch, it gets fixed and the exploit no longer works. Whereas with a virus there is no fix, and they continue to work.*

*Herein lies the problem of the whole virus community. \*THE VENDOR HAS NEVER RELEASED FIX\*. You have been so conditioned by having the vendor never release a fix that you have come to believe there is no fix.*

*Can you imagine if tomorrow Microsoft released a patch for Office that disabled any dangerous functions in macros by default and not only that but executed the dumb things in a separate compartment where they could do no damage? You can't? I understand. It would mean the end of most macro viruses as we know it. Just like that. A \*FIX\* to a problem [Levy, 1999].*

We asked a number of antivirus researchers for their opinions on the full disclosure of viruses on these mailing lists. These are representative responses:

*According to my opinion, irresponsible, unethical and damaging. Stimulates irresponsible persons spreading and creating viruses. I cannot find any truly positive impact. [Helenius, 1999]*

*Moreover, my experience with Bugtraq/NTBugtraq guys tells me they are pretty damn close to becoming cyberterrorists - fighters for what they see as a just cause which will not stop at \_anything\_ to achieve their goals. For them, ends justify the means... [Gryaznov, 1999]*

*Irresponsible and unethical. The supposed security experts have no clue about how virus writers behave. [Kuo, 1999]*

*Stupid, and potentially more harmful than useful. [Skulason, 1999]*

*Unethical, unhelpful and harmful to the computer community [Cluley, 1999]*

So, influential members of the security world in a position to contain the dissemination of viral code believe that this is a necessary evil; the anti-virus industry believes this publication is unethical and harmful. How did two groups of people with essentially the same aim end up in this position?

## **Why We Are Where We Are**

Until mid-1990s security bugs and exploits were primarily shared amongst individuals in the relatively tightly knit hacking community, via private mailing lists, closed FTP sites and invisible IRC channels. In 1994 a new model began to emerge alongside increased global Internet connectivity: hackers who once shared with a relatively small group began making tools and exploits more widely available after they became bored with the tools, or the exploit/vulnerability had reached saturation [Gordon, 1994, Berg, 1996]. Tools and exploits that were once kept relatively quiet began to emerge on Usenet with alarming frequency. Now, many exploits and hacking tools are publicly and readily available via FTP sites and on the

WWW as well; anyone armed with a Web browser can gather a considerable arsenal of exploits [NCIS, 1999 ].

Whereas hacking systems have been in existence for some time, the virus exchange phenomenon was a relative latecomer to the underground scene. The first virus exchange (vX) bulletin boards appeared in the late 1980s. They were similar in most respects to mainstream bulletin board systems; publicized by word of mouth, electronic mail and advertising on other similar systems. Many early systems required the user to upload viruses in order to download other viruses; some did not. Within roughly a three-year period from 1990-1993, the operators and users of such systems had formed a relatively small but tightly knit community; the formation of organized networks using the *FidoNet* infrastructure followed. These were still private systems. [Gordon, 1994].

As young virus writers progressed to college, access to Internet facilities became available to them. University ftp sites were sometimes used as virus distribution sites. The upload/download ratio disappeared for the most part with the availability of this Internet-based virus distribution. As interest in viruses grew, the abilities and resources of the virus writers and distributors grew. As the skills of the virus distributors evolved, they developed automated distribution programs (bots) to facilitate the distribution of viruses. By contacting one of the servers via electronic mail, or by asking for the files via direct client-to-client protocols, a user could retrieve viruses via the Internet “bot” with relative anonymity. These were still relatively controlled, private transactions and the viruses were not available to the general public [Gordon, 1994]. However, now, there are Usenet news groups dedicated to the publication of virus source code, and compiled viruses and source code are freely available on the World Wide Web to the general public. Various “underground” newsgroups and numerous FTP and WWW sites exist where various people, including administrators, can (and do) obtain viruses and information about viruses, including full analysis and source code [Martinez, 1999; Stroh, 1999].

Thus, in many ways, there are distinct similarities between the hacking and virus writing communities. In the former case, exploits have been historically quietly shared and then eventually widebanded; in the latter, we have observed a shift away from “closed group” sharing (or more correctly, exchange) to open availability on the World Wide Web. However, here the similarities end. In the anti-virus world, there is a clear delineation between the “white hats” (those working to prevent the spread of viruses) and the

“black hats” (those making the problem worse). In the security world, it is not always as clear who is who.

Within the security community few would consider *Bugtraq* and *NTBugtraq* to be “black hat” lists. However, what about the people who create exploits and send them in to these lists? In virus terms, these people might be analogous to virus writers. Thus, to a virus researcher, by their actions they would be “black hats”. Within the security world, this is not necessarily the case. In many ways, the development and public dissemination of exploits has helped strengthen the security of systems [Howard 1997, Glass 1999]. Additionally, one can observe a strong correlation between the publication of vulnerability within a public forum and a subsequent fix by the vendor [Securityfocus, 1999]. Thus, in the security world, the writers of the exploits are not necessarily perceived to be “black hats”.

The implications of this difference color the interpretation of information disclosure within the industry and between industries. When either industry measures the other by its own yardstick, it comes up short. To some in the security world, the anti-virus industry is self-serving and closed; it has failed to provide a *real* fix for the users. At best it has supplied a band-aid which needs to be replaced every week. Conversely, participants on full-disclosure security lists appear to be “black hats”, as they either actively take part in or allow the publication of virus source, something not done within the general anti-virus worldview.

This leads to the crux of the problem: both groups have strongly entrenched positions. However, neither has concrete data to back up its position. One key to improving the current situation is to try and quantify the result of publishing virus code.

### **Finding the Facts**

There are a number of interesting avenues of research that could help enable us to present fact as opposed to opinion when attempting to influence disclosure positions.

First, one could examine a virus whose source code has been widely distributed on Usenet. For example, in the case of Melissa, were there an unusual number of variants found following the publication of the source code? Furthermore, of these variants, was there a significantly higher percentage actually observed “In the Wild”?

It would be interesting to know what administrators and others who downloaded the source to Melissa actually did with it: was the sample used for product testing? Testing of a sendmail patch? Creation of a variant?

Similar analysis could be carried out on a number of different viruses.

One of the problems with this approach is the small data-set; the number of viruses posted in source form on “main-stream” mailing lists is vanishingly small compared to the total number of known viruses. The small number of data points will therefore make the potential for error large.

The situation is made more complex by the presence of other variables. These confounding factors include people gathering viruses from mailing lists when other sources of information “dry up” due to law enforcement attention, school vacations, or historical events. A virus may be posted to a list *because* it is spreading or has gained a lot of public attention. The difficulty involved in separating cause from effect is large. However, if the dissemination of virus source on these lists is really causing a significant problem, it seems likely that some positive correlation will be shown.

Similar research has already been carried out by [Gordon, 1993], with the difference that this research examined the impact from those viruses placed upon vX Bulletin Board Systems. No positive correlation was demonstrated; very few of these “for distribution” viruses were found spreading in user populations. However, the Internet has added new dimensions to the problem. “Publicly available” viruses now have a much larger potential audience; there are increases in anonymity which may lead to people being more comfortable obtaining viruses; there has been a tacit approval/lack of societal sanction related to obtaining viruses to be used in the course of one’s job. Further research exploring *any* correlation between publication of viral source on the Internet and those viruses “in the wild” would be useful. In the case of publication of source, we note that this is not to be confused with the *unlabelled* publication of infected documents or files - that is, the publication of a virus without clearly labeling it as such.

There is also the question of legal liability. If one were to publish a virus within a moderated list, clearly labeling the sample or source code, is there any question of liability, either by the publisher or the list moderator who chose to allow the publication? Some have stated the authors of the virus or the publishers should be held liable [FitzGerald, 1999]; however, this is a complex issue which involves both an understanding of not only the technologies involved, but of the intricacies of international law as well.

Within the United States, computer viruses are not expressly prohibited under many statutes related to unauthorized computer use, access or manipulation of computers or computerized data [Rasch, 1996]; even if they were, some liabilities related to network traffic in general are impossible to resolve due to jurisdictional aspects [Cook, 1993]. While these issues may be more resolvable when dealt with as social problems than as criminal or civil litigations, criminal, civil and third party liabilities in cases of virus infection may provide some recourse [Gordon, 1994; Loundy, 1998]. However, a *direct* action (i.e., active placement of the virus) rather than mere publication is the crux of all of these arguments. The consensus appears to be that *potential* for abuse of information is not sufficient cause for censorship of the information. Research into viruses as “free speech” is a much-discussed topic; despite finding studies related to current policy and regulation regarding viruses in general [Tribe, 1991; Emilian, 1998; Novell, 1998; Thomas, 1993], we were unable to find any definite conclusion as to whether or not the *publication* of virus source code can be a legally actionable. Research documenting this type of case law could prove useful.

We have explained ways in which virus source code is now widely publicly available. Given this availability, it is not uncommon for IS administrators to obtain and examine such code in the course of their job-related tasks [Tirado, 1993; Melka, 1993; Anonymous, 1999a]. For this reason, we should not be surprised that many IS people seem to think such public availability is a “good thing” or at worst “a negligible harm”.

Despite some published research demonstrating that such public availability is less than useful (and may be harmful) [Schibsted, 1998; Gordon and Ford, 1995; Gordon, 1996], the media has often taken the position that such publication is a “good thing” [Louderback, 1999; Silverstone, 1999].

How reflective are these positions of the Information Security professionals who actually use the mailing lists to help with their daily tasks? We made initial inquiries on the usefulness of such publication; generally, respondents indicated they believed such publication to be, at worst, negligibly harmful and at best, extremely helpful. However, the responses we gathered were insufficient to allow for a statistically meaningful result in either case; therefore, further research is warranted. For example, it might prove useful to poll IS managers who are favorable toward publication of the source code of new virus types, asking them something like “if it makes it 10x more likely the virus will be found spreading In the Wild, do you still want it published on security mailing lists?”

One problem is that we have little idea of the view of the “real” majority, as most of the public communication regarding this issue is dominated by only a few loud voices. Thus, as we are trying to develop mutually beneficial sharing models, it is important for every informed person to share their views and make their voice heard - this type of interaction is critical in avoiding pluralistic ignorance [Shaw, 1981]. It is important to realize that non-participation could easily be seen as tacit agreement to a particular worldview; to quote Edmund Burke: “All that is necessary for evil to triumph is for good men to do nothing”.<sup>1</sup>

### **Irreconcilable Differences?**

So far, we have explored the differences in position between the anti-virus community and parts of the security community, and pointed to problems in reconciling these differences; we have yet to discuss why resolution of these differences is important. In this section, we shall briefly touch upon the risks of allowing these differences to remain unresolved.

The first outcome has a direct impact on the user. If the anti-virus position that publication of these viruses leads to direct harm to the user is correct, the continued publication of viral source on mainstream security lists leads to *direct* user harm.

Conversely, if the full disclosure position is true, and publication is a net win for users, lack of cooperation in this endeavor by the anti-virus industry could prevent helpful information from filtering down to administrators. In the worst case, the anti-virus industry could even support legislation to outlaw this publication, making a useful tool illegal. Some have already suggested this type of restriction might be useful [Solomon, 1992; Tippet, 1993].

The very fact that the two groups have such diametrically-opposed positions does not necessitate internecine war. Consider the relationship between *CERT* (very limited disclosure) and *Bugtraq*: despite taking quite different approaches in terms of information disclosure, *CERT* lists both *Bugtraq* and *NTBugtraq* as important resources for security professionals. The lists are cited along with Virus-L, comp.risks, alt.security, comp.security.misc and others, accompanied by a disclaimer stating they do not necessarily endorse

---

<sup>1</sup> Although this is the “famous” part of the quote, it is somewhat out of context. The entire quote reads “The true danger is when liberty is nibbled away, for expedience, and by parts...the only thing necessary for evil to triumph is for good men to do nothing.”

all of the lists [CERT, 1999]. This is because both groups recognise the complex issues surrounding full-disclosure. Unfortunately, due to the strong feelings regarding sharing of virus code, full-disclosure security lists appear to virus researchers to be “black hat” lists - that is, lists to be rightfully combated. Conversely, the apparent failure of the anti-virus industry to produce a solution to the virus problem cries out to those inclined toward full-disclosure as an area to apply continued pressure. This exact sentiment was echoed by Cooper in a preceding section.

A less direct result of continued hostility could be poorer integration between security products and anti-virus products. This lack of synergy simply means that users are likely to miss out on many of the benefits brought by cross-disciplinary cooperation. Each group has technology and knowledge that could be helpful to the other; it would be unfortunate if reduced cooperation simply helps perpetuate global “security” problems.

[Howard, 1997] recommends that response teams which deal with general computer security issues reexamine their policies toward the release of vulnerability information with the objective of seeing the degree to which more disclosure would benefit the Internet community. Similarly, we must continually evaluate the impact of sharing of virus samples and code both inside and outside the mainstream anti-virus community to see whether our actions provide a strong benefit to the global Internet community.

### **Until We Have The Facts...**

The proposed research will take time; in the interim, we suggest that all parties involved “play nice”, else the negative outcomes outlined above will dominate. Note that it is possible to cooperate despite significant differences in position on the issue of disclosure: as stated earlier, *CERT* and *Bugtraq* certainly do not share the same stance, yet there is no public animosity between these two groups. This is not likely to simply be due to fact that these groups are composed of “nice guys” (although they are!). Rather, it is due to the acceptance of the fact that each has a good argument for their chosen direction, as well as the fact that there is a good and productive flow of information between them. The peaceful co-existence of *Bugtraq* and *CERT* illustrates the benefits of engaging in productive dialogue.

Research by [Larson, 1997, Stasser, Taylor & Hanna, 1989] supports the statement that *if communications are disrupted early on, it is possible that critical information sharing may never take place*. We have observed this to be true in private communication within the anti-virus world. In some cases,

sample distribution has been deliberately held back, leading to thousands of avoidable infections [Anonymous, 1996; Anonymous, 1999b; Pecelj, 1999]: this issue is not theoretical... it has very real consequences for users.

We should remember that understanding the problem at hand, understanding the requirements for an acceptable solution, and assessing the consequences (positive and negative) that could be associated with all of the alternatives are critical functions to developing solid solutions to problems. The extent to which members of any problem-solving group are able to *successfully* perform these functions is likely to be influenced by the group's interaction style or manner in which members conduct themselves in dealing with one another. [Cooke & Lafferty, 1988; McGrath 1984, Cooke & Szumal, 1994]. Thus, until we can successfully gather data regarding the net result of the publication of the source to new virus threats (like Melissa), it seems best that we all work toward keeping meaningful dialogue open.

## **Conclusion**

As the role of anti-virus software continues to move toward more general "security" solutions, we expect an increase in the integration of anti-virus software with security solutions. However, we have observed that some important participants in these two communities have radically different information sharing models. These differences have arisen due to the ways in which each community has developed, and because of differences in the types of information shared. In each community, there exists a significant number of members who hold views which are diametrically opposed to their counterparts in the other.

Disagreements between members of the community can be traced back not to simple personal animosity, but to differences in these information sharing models. In particular, it is important to realize that breakdown of communication is not merely a temporary "flame" on a mailing list, but more deeply rooted, arising from the different nature of the populations. This in turn leads to a potential significant increase in the risk faced by computer users in general.

One of the complications faced is that neither side has convincing evidence of the impact of their actions. Thus, we propose research be carried out which attempts to quantify the impact of "full" disclosure with respect to viruses. We note that such research is likely to be prolonged, given the difficulties in data gathering. However, we note that without such attempt at

objective research, the differences in paradigms look set to remain, hampering communication and cooperation.

We suggest that the time is ripe for a re-examination of the issues concerning information sharing within both these communities. How much information to divulge, especially with respect to autonomous entities like computer viruses is a delicate balancing act: too little information distributed means that users may not be able to take steps to protect themselves; too much, and the problem may be made significantly worse.

Furthermore, and most importantly, we propose that a concerted effort is made to encourage synergy and communication between groups is made. There is a pressing need to the merger of desktop security and anti-virus software to be hastened; viruses, non-viral malicious code and simple hack attempts have much in common; only by taking a broad-spectrum approach can we ever hope to optimize protection for the user.

## References

Anonymous, 1996. Private electronic communication. Used with permission.

Anonymous, 1999a. Private electronic communication. Used with permission.

Anonymous, 1999b. Private electronic communication. Used with permission.

Berg, A. 1999. *Net App Opens Doors for Hackers*. LANTIMES [online] <http://www.lantimes.com/96aug/608b016a.html>.

CERT, 1999. [online] [http://www.cert.org/nav/other\\_sources/usenet.html](http://www.cert.org/nav/other_sources/usenet.html).

Chess, D. 1999. Private electronic communication. Used with permission.

Cluley, G. 1999. Private Electronic Communication. Used with permission.

Cohen, F. 1996. *Internet Holes - Automated Attack and Defense*. Management Analytics. [online] <http://all.net/journal/netsec/9601.html>

Cook, W. 1993. *Network Traffic Liability*. Invited Op-ed. American Association for the Advancement of Science. Irving, California.

Cooke, R. A., & Lafferty, J. C. 1988. *Group Styles Inventory*. Plymouth, MI: Human Synergistics.

- Cooke, R. & Szumal, J. 1994. *The impact of group interaction styles on problem-solving effectiveness*. Journal of Applied Behavioral Science, Dec94, Vol. 30 Issue 4, p415.
- Cooper, R. 1999. Private Electronic Communication. Used with permission.
- Cowan, C., Pu C., Maier, D., Walpole, J. Bakke, P., Beattie, S., Grier, A., Hinton, H., Wagle, P, and Zhang. Q. 1998. *StackGuard: Automatic Adaptive Detection and Prevention of Buffer -Overflow Attacks*. From the Proceedings of the Seventh USENIX Security Symposium. San Antonio, Texas. [online]  
[http://www.cse.org.edu/DISC/projects/immunix/stackguard\\_usenix98.ps.gz](http://www.cse.org.edu/DISC/projects/immunix/stackguard_usenix98.ps.gz)
- Defoe, D. 1999. *Melissa: Bad News, Good News*. [online]  
[http://www.infosecnews.com/scmagazine/1999\\_05/cover/cover.html](http://www.infosecnews.com/scmagazine/1999_05/cover/cover.html)
- Denning, D. 1999. *Information Warfare and Security*. p375. Addison Wesley Publications.
- Emilian, J. 1998. *Too Hot to Handle! The Drafters of Proposed Article 2B Drop the Electronic Viruses Section*. Submitted in partial fulfillment of the thesis requirement of High Technology Law Degree. Suffolk University Law School. Boston, Massachusetts.
- FitzGerald, N. 1999. Private electronic communication. Used with permission.
- Garfinkel, S. & Spafford, E. 199X. *Practical Unix Security*. p895. O'Reilly Publications
- Glass, B. 1999. *Should security flaws be posted?* Technology. [online]  
<http://www.msnbc.com/news/281515.asp>
- Gordon, 1993. *Circular Time-Line Model for Evaluating the Impact of Virus-Exchange BBS*. Presentation for the 6<sup>th</sup> International Computer Security and Virus Prevention Conference. DPMA Financial Industries Chapter. New York City, New York.
- Gordon, S. 1994. *Technologically Enabled Crime: Shifting Paradigms for the Year 2000*. Computers and Security. Elsevier Science Publications.
- Gordon, S. 1996. *Real-World Anti-Virus Product Reviews and Evaluation: The Current State of Affairs*. From the Proceedings of the 19<sup>th</sup> National Information Systems Security Conference. NIST/NSA. Baltimore, MD.
- Gordon, S. 1998. *The Worm Has Turned*. *Virus Bulletin*. July issue. pp10-12.

- Gordon, S. & Ford, R. 1995. *Real-World Anti-Virus Product Reviews and Evaluation*. From the Proceedings of Security on the I-WAY. NCSA. Crystal City, Virginia.
- Gryaznov, D. 1999. Private electronic communication. Used with permission.
- Hart, 1999. *Piracy on the IT*. National Criminal Intelligence Service. [online] <http://www.ncis.co.uk>
- Helenius, M. 1999. Private electronic communication. Used with permission.
- Howard, J. 1997. *An Analysis Of Security Incidents On The Internet. 1989 - 1995*. A dissertation submitted to the graduate school in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Engineering and Public Policy. Carnegie-Mellon University. Pittsburgh, Pennsylvania
- Kuo, J. 1999. private electronic communication. Used with permission.
- Larson Jr., James R. *Modeling the entry of shared and unshared information into group discussion*. Small Group Research, Aug97, Vol. 28 Issue 3, p454.
- Levy, E. 1999. Private electronic communication. Used with permission.
- Louderback, J. 1999. *Viruses' Brave New World*. Commentary. [online] <http://www.zdnet.com/zdnn.stories/comment/0,5859,2288121,00.html>
- Loundy, D. 1998. *Computer Information Systems Law and System Operator Liability*. Seattle University Law Review, Volume 21, Number 4.
- Martinez, M. 1999. *The 'Why' of Viruses*. [online] <http://abcnews.go.com/sections/tech/DailyNews/viruswriters990409.html>
- Melka, P. 1993. Computer Underground Digest. Ed. Jim Thomas & Gordon Meyer. July 11. Volume 5, Issue 51. ISSN 1004-942X.
- McGrath, J. 1984. *Groups: Interaction and performance*. Englewood Cliffs, NJ: Prentice-Hall.
- Novell, J. 1998. *Internet Abuse: A survey of the current state of regulation, and a call for change*. Villanova Information Law Chronicle. Villanova University. [online] <http://www.vcilp.org/vill.info.1.chron/articles/novell.html>
- Patrizo, A. 1999. *New Viruses Send Data Over Internet*. Techweb Technology News. [online] <http://www.techweb.com/wire/story/TWB19990205S0011>

- Pecelj, D. 1999. Private electronic communication. Used with permission.
- Rasch, M. 1996. *Criminal Law and the Internet. The Internet and Business: A Lawyer's Guide to the Emerging Legal Issues*. Computer Law Association.
- Schibsted, E. 1998. *At work with the IBM Antivirus Expert*. Forbes ASAP.
- Securityfocus. 1999. [online] <http://www.securityfocus.com>
- Shaw, M., 1981. *Group dynamics: The psychology of small group behavior*. New York: McGraw-Hill.
- Silverstone, S. 1999. *Newsgroups reflect virus' scope, speed*. Computing. ZDNN Tech News Now. March 27<sup>th</sup>.
- Skulason, F. 1999. Private electronic communication. Used with permission.
- Solomon, A. 1992. Presentation for the National Computer Security Association. Crystal City, Virginia.
- Stasser, G., Taylor, L. A., & Hanna, C. (1989). *Information sampling in structured and unstructured discussions of three- and six-person groups*. Journal of Personality and Social Psychology, 57, 67-78.
- Stroh, M. 1999. *What does a virus writer look like?* Technology. The Baltimore Sun. Friday, April 9<sup>th</sup>.
- Sullivan, B. 1999. *Privacy Flaw offers digital clues in hunt for 'Melissa' author*. [online] <http://www.msnbc.com/news/25803.asp>
- Thomas, J. 1993. Computer Underground Digest. ed. Jim Thomas & Gordon Meyer. July 11. Volume 5, Issue 51. ISSN 1004-942X.
- Tirado, F. 1993. Computer Underground Digest. ed. Jim Thomas & Gordon Meyer. July 11. Volume 5, Issue 51. ISSN 1004-942X.
- Tippett, P. 1993. *Hearings Before the Subcommittee on Telecommunications and Finance of the Committee on Energy and Commerce*. U.S. Government Printing Office.
- Tribe, L. 1991. *The Constitution in Cyberspace: Law and Liberty Beyond the Electronic Frontier*. [online] <http://www.epic.org/free-speech/tribe.html>