

Security in Cyberspace: Combatting Distributed Denial of Service Attacks

Jennifer A. Chandler*

233	INTRODUCTION
234	1. THE GENERAL PROBLEM OF CYBER SECURITY
236	2. DISTRIBUTED DENIAL OF SERVICE ATTACKS (“DDOS” ATTACKS)
240	3. PREVENTION OF DDOS ATTACKS
240	3.1. <i>Introduction</i>
241	3.2. <i>The Victim of a DDOS Attack</i>
241	3.3. <i>The “Masterminds”</i>
243	3.4. <i>Internet Service Providers</i>
244	3.5. <i>Internet-Connected Computer Users</i>
248	3.6. <i>Software Developers</i>
255	4. LIABILITY IN NEGLIGENCE FOR UNREASONABLY INSECURE SOFTWARE
256	4.1. <i>Duty of Care</i>
259	4.2. <i>The Harm of DDOS Attacks: Pure Economic Loss?</i>
261	CONCLUSION

Copyright © 2004 by Jennifer A. Chandler

* Assistant Professor of Law, University of Ottawa. The author thanks the Centre for Innovation Law and Policy for financial support for the writing of this article. The author also thanks Paul Zatychech and Dean Bruce Feldthusen (of the University of Ottawa Faculty of Law) for most valuable discussions and suggestions, as well as Dina Mashayekhi and William David for excellent research assistance. Errors and omissions remain the responsibility of the author.

Security in Cyberspace: Combatting Distributed Denial of Service Attacks

Jennifer A. Chandler

INTRODUCTION

THE POOR STATE OF CYBER SECURITY has finally begun to attract broad attention and concern outside the community of computer security experts. This is a welcome development since, whether or not the worst fears of a “digital Pearl Harbor”¹ are realistic, it is clear that cyber attacks impose heavy costs and that the rate of attack is increasing.² Security vulnerabilities continue to be discovered and disclosed in widely-deployed software at a great rate. Yet, internet-connected computer users generally fail to take basic steps to patch the vulnerabilities in their software and to safeguard their systems from malicious code. Despite the large accumulated losses and warnings of more serious dangers, the internet remains a place of cyber insecurity.

Cyber security is not a single problem, but rather a group of very different problems involving various sets of threats, targets and costs. As a result, legal policy analysis must begin by identifying the particular problem to be considered. This paper focuses on a form of cyber attack known as a distributed denial of service attack (“DDOS”). DDOS attacks are interesting from the legal perspective because there is essentially nothing that a victim can do to protect itself, and because the actual perpetrators of the attacks are nearly impossible to trace at present. As a result, if the law wishes to discourage DDOS attacks, some other object of legal pressure must be chosen.

One root cause of DDOS attacks is insecure software. Insecurities in software programs that have been widely deployed on internet-connected computers are exploited in order to conscript computers into an army of “zombies” that is later used by an attacker to launch the DDOS attack. A properly functioning software market might be expected to generate the proper balance of price and

-
1. Declan McCullagh, “Cyberterror and Professional Paranoiacs” *CNet News.com* (21 March 2003), <<http://news.com.com/2102-1071-993594.html>>, citing Richard Clarke.
 2. CERT/CC Statistics 1988–2003: <http://www.cert.org/stats/cert_stats.html>. The CERT Statistics reveal that the number of incidents and vulnerabilities reported annually has exploded between 1988 and 2003.

quality characteristics (including the level of security). However, for reasons that will be discussed in greater detail later, such as the existence of “insecurity externalities,” it is possible that investment in security is presently inadequate.

This paper suggests that DDOS attacks could be reduced by improving software security, and that a promising way to improve software security is for the targets of DDOS attacks to sue the market-dominant vendors of critical software for creating an unreasonable risk of harm from attack by third parties. The target of a DDOS attack is the best suited plaintiff in a lawsuit against the vendor of insecure software. These plaintiffs are not open to charges of contributory negligence as there is essentially nothing they can do to protect themselves. They suffer the kind of concentrated loss that would make litigation attractive. In addition, they do not face the obstacle of contractual disclaimers and limitations of liability within the software licence agreements that exist between software vendors and the owners of the insecure computers used to launch a DDOS attack.

It is suggested in this paper that, like a landlord or occupier who is required to ensure the safety of those within a physical space under the landlord’s control, a near-monopolist vendor of software that defines the structure of cyberspace must also ensure the safety of those in that virtual space.

Part 1 of this paper will outline the general problem of cyber security. Part 2 will discuss the particular problem of distributed denial of service (“DDOS”) attacks. Part 3 will address the prevention of DDOS attacks, discussing measures that might be taken at the level of the software vendor, the internet-connected computer user, the Internet Service Provider “ISP”, the perpetrator of the DDOS attack, and the victim of the DDOS attack. Finally, Part 4 will explore the possibility of holding vendors of unreasonably insecure software liable in negligence for creating an unreasonable risk of harm to the victim of a DDOS attack.

*

1. THE GENERAL PROBLEM OF CYBER SECURITY

MANY DIFFERENT PROBLEMS are customarily grouped within the subject of “computer network security.” While they share certain basic characteristics (e.g. the deliberate exploitation of a vulnerability in an information system for improper purposes), other characteristics that are legally important are not necessarily shared. The attackers, targets, victims, harms and possible defensive measures are legally relevant characteristics, but they vary among the forms of cyber attacks. As a result, it is likely best not to attempt to make legal policy recommendations aimed at cyber security generally. For example, the failure of a business to safeguard the sensitive personal information of customers might be met with various legal responses directed at the enterprise with lax security such as legislated security standards,³ mandatory public disclosure of security

3. Legislation has been enacted in the United States to cover health-related information (see the *Health Insurance Portability and Accountability Act of 1996*, Pub. L. No. 104-191, 110 Stat. 1936), <http://frwebgate.access.gpo.gov/cgi-bin/getdoc.cgi?dbname=104_cong_bills&docid=f:h3103enr.txt.pdf>), and financial information (see the *Gramm-Leach-Bliley Act*, Pub. L. No. 106-102, <http://frwebgate.access.gpo.gov/cgi-bin/getdoc.cgi?dbname=106_cong_bills&docid=f:s900enr.txt.pdf>).

breaches,⁴ mandatory reporting of security status,⁵ or negligence lawsuits. Different responses may be required in the context of a distributed denial of service attack, in which the harm is essentially unavoidable by the victim. As a result, at least in the beginning, it is necessary to address the various facets of the “cyber security” problem separately.

At a basic level, computer network security is understood to consist of three requirements: data secrecy, data integrity and computer asset availability.⁶ Stallings suggests another useful analytical structure, namely that cyber attacks on the security of a computer system or network are best conceived of as problems in the flow of information from a source to a user. The attacks can thus be reduced to four general categories of problem in the flow of information: interruption, interception, modification and fabrication.⁷ An interruption in the flow of data arises when part of the system is destroyed or disabled. An interception of data destroys confidentiality, occurring when an unauthorized party gains access to data. Modification of data is an attack on its integrity, and occurs when an unauthorized party gains access to data and tampers with it. Fabrication of data occurs when an unauthorized party inserts counterfeit data into a system.

This type of analytical structure is most useful for understanding the nature of cyber security as a topic, but it is too abstract for the purposes of legal analysis. In particular, laws must be designed in light of the origin of an attack, the motive for an attack, the form of an attack, the target, and the type of damage done. For example, the origin of an attack may be an outsider or a trusted insider, and it may be domestic or foreign. Attacks may be conducted for reasons of financial gain, espionage, electronic warfare, personal vendetta, political or ideological conviction, the thrill of the challenge, or the acquisition of boasting rights. Attacks may be conducted solely via the internet, through the recruitment of a trusted insider, or in the form of physical attacks on a component of the network.⁸ Possible targets include anything connected to the network, including home computers, web sites, confidential information and communications on networked computers, and control systems for critical infrastructures such as power distribution grids. The damage done may consist of interrupted or degraded network services, lost or corrupted data, loss of confidentiality, and, in extreme cases, property damage and personal injury.

4. *California Civil Code Section 1798.82*, added by California Bill Number SB 1386 (operative as of 1 July 2003), <<http://www.leginfo.ca.gov/cgi-bin/waisgate?WAIISdocID=0280642085+6+0+0&WAIISaction=retrieve>>, requires the holders of personal information about a California resident to disclose to the California resident breaches of the security of the personal data in certain circumstances.

5. Declan McCullagh, “PC Security Audits for Businesses?” *CNet News.com* (6 November 2003), <<http://news.com.com/2100-7355-5103564.html>>.

6. William Stallings, *Data and Computer Communications*, 5th ed. (Upper Saddle River, NJ: Prentice-Hall, 1997) at 624. Secrecy refers to the need to protect the confidentiality of data, ensuring that only authorized parties may read it. Integrity means that data cannot be modified (by adding, deleting, or changing data) except by authorized parties. Availability means that computer assets must remain available to authorized parties.

7. *Ibid.* at 625.

8. U.S., Computer Science and Telecommunications Board, “Cybersecurity, Today and Tomorrow, Pay Now or Pay Later” (Washington: National Academies, 2002), <<http://books.nap.edu/html/cybersecurity/>> under *Causes of System and Network Problems*.

This paper focuses on one particular form of attack, namely the DDOS attack. This form of attack was chosen as the focus of this paper because it presents certain characteristics that are interesting from the legal perspective. As will be discussed further in the following section, there are several types of parties involved in some way in a DDOS attack, including the perpetrator(s) of the attack, software developers, computer users, internet service providers and the target of the attack. Laws aimed at the prevention of DDOS attacks may be most effective only against parties in the middle of this chain due to the difficulty of tracing and punishing the perpetrators and the lack of effective self-protective measures available to targets of such attacks. In addition to the interesting legal problem of the appropriate target of a law attempting to combat DDOS attacks, it is also interesting to consider the appropriate legal response in light of potential imperfections in the market for software. For example, an economic externality problem may lie at the root of DDOS attacks, given that those whose under-investment in security creates the risk of DDOS attacks (i.e., the eventual "zombies") will not suffer from the harm of those DDOS attacks. Consequently, they do not face adequate incentives to invest in security.

★

2. DISTRIBUTED DENIAL OF SERVICE ATTACKS ("DDOS" ATTACKS)

A DENIAL OF SERVICE ATTACK is an attack that seeks to disable the target so that it no longer is able to offer the services it normally provides. In the usual internet scenario, a denial of service attack is an attack in which a server is deliberately sent a large volume of communications traffic that overwhelms it and causes it to crash.⁹

A DDOS attack is one in which the attack is launched against one target simultaneously from a large number of sources. DDOS attacks often make use of armies of computers that have been previously infected with malicious code that enables an outsider to use them to launch the attack. These infected computers are referred to as "zombies," "agents," "slaves," or "bots."¹⁰

Similar losses of service may result at various points in the internet, not from a deliberate DDOS attack against a particular target, but because of a computer virus or worm epidemic, as servers struggle to deal with the increased traffic caused by the propagation of the infection. Some servers simply disconnect from the internet during an intense epidemic, effectively imposing denial of serv-

9. Kevin J. Connolly, *Law of Internet Security and Privacy* (New York: Aspen, 2003) at 48. Allen Householder et al., "Managing the Threat of Denial-of-Service Attacks" v. 10.0, CERT Coordination Center, (October 2001, Carnegie Mellon University) at 21–22, <http://www.cert.org/archive/pdf/Managing_DoS.pdf>. Householder et al. note that there are three general categories of DDOS attacks: bandwidth attacks, protocol attacks and software vulnerability attacks. Bandwidth, or throughput attacks are simple attempts to consume resources such as network bandwidth or equipment throughput capacity. This is done by sending massive numbers of data packets so that legitimate traffic slows down as the target system struggles to process the incoming traffic. Protocol attacks take advantage of the normal behaviour of internet communication protocols such as TCP, UDP and ICMP. For example, a "SYN flood" consists of a flood of TCP SYN packets, perceived by the victim as attempts to establish a connection. Software vulnerability attacks exploit vulnerabilities in network software by, for example, sending malformed packets that cannot be properly handled.

10. Householder, *ibid.* at 22. They will be referred to in this paper as "zombies."

ice upon themselves.¹¹ For example, in August 2003 the MSBlaster worm shut down the Maryland Motor Vehicle Administration for a day.¹² Another worm, known as “Welchi,” “Welchia,” or “Nachi,” which was actually intended to protect computers from MSBlaster,¹³ brought down the Air Canada check-in system, infiltrated unclassified computers on the U.S. navy intranet, and briefly shut down the train signaling system at CSX Corp., the third-largest U.S. railroad operator.¹⁴ The infection resulted in train delays and cancellations as signals went dead.

The mastermind of a DDOS attack may spread the software used to launch a DDOS attack to vulnerable computers within malicious code such as viruses, worms or Trojan horse programs, or may simply take advantage of DDOS attack tools previously installed on infected computers by others. Various automated tools, such as programs to scan for vulnerable systems, to attempt exploitation of those found, and to compile a list of the compromised systems, have greatly eased the work in assembling an army of “zombies.”¹⁵ The network of infected computers can later be activated remotely by the mastermind, who can encrypt his or her communications with the “zombies” to prevent detection.¹⁶ Attacks launched from the infected computers often forge false addresses of origin to make it difficult for the target to block the illegitimate traffic or to trace the attack to the infected computers.¹⁷

The internet’s first big wave of DDOS attacks took place in 2000, and temporarily disabled Yahoo!, Amazon.com, CNN, E*Trade, ZDNet, Buy.com, Excite and eBay.¹⁸ Apart from these prominent casualties of DDOS attacks, DDOS attacks appear to be being continually launched at a fairly high rate. A CERT Coordination Center¹⁹ paper on denial of service attacks cites the astonishing results of a study conducted by researchers at the Cooperative Association for Internet Data Analysis (“CAIDA”). They observed nearly 13,000 attacks on over 5,000 different internet hosts belonging to more than 2,000 distinct organ-

-
11. Kevin J. Houle & George M. Weaver, “Trends in Denial of Service Attack Technology” v. 1.0, CERT Coordination Center, (October 2001, Carnegie Mellon University) at 18–19, <http://www.cert.org/archive/pdf/DoS_trends.pdf>.
 12. Charles Duhigg “Strong Attackers, Weak Software” *The Washington Post* (21 August 2003) E01, <<http://www.washingtonpost.com/ac2/wp-dyn/A23020-2003Aug20?language=printer>>.
 13. The worm caused infected computers to download the Microsoft patch that would protect them from the MSBlaster worm. See CERT, “CERT/CC Current Activity Page” with reference to the Welchia worm, added 18 August 2003, <http://www.cert.org/current/current_activity.html#welchia>.
 14. Duhigg, *supra* note 12.
 15. Houle & Weaver, *supra* note 11 at 10.
 16. Richard D. Pethia, “Computer Security” 9 March 2000, Testimony before the (United States) Committee on Government Reform, Subcommittee on Government Management, Information and Technology, <www.cert.org/congressional_testimony/Pethia_testimony_Mar9.html>. See also Houle & Weaver, *supra* note 11 at 4.
 17. Pethia, *ibid*.
 18. Ed Skoudis, “InfoSec’s Worst Nightmares” Information Security Magazine, (November 2002) <<http://www.infosecurymag.techtarget.com/2002/nov/nightmares.shtml#1d>> at DDOS Attacks (2000). See also Stephen E. Henderson & Matthew E. Yarbrough, “The Internet and Cyberspace: Suing the Insecure?: A Duty of Care in Cyberspace” (2002) 32 N.M.L. Rev. 11. At page 11, Henderson and Yarbrough list CNN, Yahoo!, Amazon.com, eBay and Dell as targets of DDOS attacks in February 2000 with estimated damages as high as \$1.7 billion.
 19. The original name of CERT was the “Computer Emergency Response Team”.

izations during a three-week period in 2001.²⁰ The method of analysis used would have picked up only DDOS attacks that used packets with spoofed source addresses, and would have missed those that did not do so.²¹

The Computer Security Institute and FBI jointly produce an annual computer crime and security survey. The 2003 report is based on responses from approximately 500 computer security practitioners in U.S. corporations, government agencies, financial and medical institutions, and universities.²² The survey revealed that 42% of respondents (the highest number since 1999) reported having been the target of a denial of service attack in the preceding 12-month period.²³ DDOS attacks are predicted to continue into the future. In 2002, the readers of InfoSecurity Magazine placed DDOS attacks and "massively distributed attacks"²⁴ within the eight biggest security threats predicted to occur between 2003 and 2008.²⁵

It is difficult to estimate the total costs of DDOS attacks. The failure to be precise or consistent about what types of losses are included in the available estimates reduces their usefulness. The harms caused to the target of a DDOS attack can include lost productivity, the costs of reaction and recovery, lost sales, negative brand impact, lost goodwill due to degradation of services, and potential legal liability for failure to meet obligations.²⁶ Not all of the respondents to the CSI/FBI 2003 survey on computer crime and security were able to quantify the losses flowing from the reported DDOS attacks, but the 251 respondents who were able to provide dollar losses reported a total of \$66.6 million in losses due to denial of service attacks during the 12-month reporting period.²⁷ This placed denial of service attacks as the second-most costly crime among survey respondents after the theft of proprietary information.²⁸

In addition, the losses of productivity related to a DDOS attack may extend more broadly than the intended target. For example, attacks on one customer of an ISP may degrade the network resources available to other customers.²⁹

20. Householder, *supra* note 9 at 2, citing CAIDA, "Inferring Internet Denial-of-Service Activity" (San Diego: University of California, 2001), <<http://www.caida.org/outreach/papers/backscatter/>>. CAIDA inferred the rate of DoS by observing the amount of "backscatter" in the unpopulated internet address space. The theory is that random spoofed source addresses on DoS attack packets will generate response traffic to the entire internet address space, including the unpopulated space. "Backscatter" analysis infers DoS activity from the amount of noise in the unpopulated space.

21. *Ibid.* at 18.

22. Robert Richardson, "Eighth Annual 2003 CSI/FBI Computer Crime and Security Survey" (San Francisco: Computer Security Institute, 2003) at 1, <<http://www.gocsi.com>>.

23. *Ibid.* at 10.

24. A massively distributed attack is an attack that goes beyond traditional DDOS to make use of distributed password cracking, distributed port and vulnerability scanning, etc. See Skoudis, *supra* note 18.

25. *Ibid.*

26. Bruce Schneier provides a useful classification for the costs of cyber attacks in general. He writes that direct losses include theft (of money, trade secrets and proprietary information, digital assets, customer information, computer resources) and productivity losses (corruption of data, expenses of recovery and continuity). Indirect losses include secondary losses (lost sales, lost competitive advantage, negative brand impact, lost goodwill) and legal liability (failure to meet contracts, failure to meet privacy obligations, illegal user activity and officer liability). See Bruce Schneier, "Risk, Complexity and Network Security" (Powerpoint Presentation, April 2001) [unpublished], <<http://www.counterpane.com/presentation1.pdf>>.

27. Richardson, *supra* note 22 at 12.

28. *Ibid.* at 4.

29. Householder, *supra* note 9 at 2.

Attacks on critical infrastructures (such as energy distribution, communications, transportation, government, emergency services, etc.) could impose physical injuries to persons or property as well as economic losses. Power grids, gas and oil distribution pipelines, water treatment and distribution systems, hydroelectric and flood control dams, oil and chemical refineries and other infrastructures have long been controlled by computer through SCADA systems (supervisory control and data acquisition systems) and other networked computer systems.³⁰ These control systems are now increasingly being connected to communications networks in order to lower costs by permitting remote maintenance, control and updating.³¹ Richard Pethia, Director of CERT, recently warned of the dangers inherent in this trend. He stated that:

[a]s critical infrastructure operators strive to improve their efficiency and lower costs, they are connecting formerly isolated systems to the Internet to facilitate remote maintenance functions and improve coordination across distributed systems. Operations of the critical infrastructures are becoming increasingly dependent on the Internet and are vulnerable to Internet based attacks.³²

A recent incident at an Ohio nuclear power plant has demonstrated the potential dangers of inattention to cyber security. On August 29, 2003, the U.S. Nuclear Regulatory Commission issued a notice entitled "Potential Vulnerability of Plant Computer Network to Worm Infection."³³ The notice was issued because of an incident on January 25, 2003 at a nuclear power plant in the United States, in which a worm infected the plant's network, taking down the "Safety Parameter Display System" for nearly five hours and disabling the plant process computer for six hours. The safe operation of the plant was not affected because the plant was off-line at the time and a redundant backup system existed in any event.³⁴ The worm entered the nuclear plant's computer network from the corporate network of the utility company running the nuclear plant due to an insecure route left open around the plant's firewall. The plant's computer network was vulnerable to attack because a software patch released about six months earlier had not been installed.³⁵

Internet security does not only concern large companies and governments in industrialized countries. As was emphasized at the recent UN Global InfoSec Conference, many developing countries are increasingly technology-dependent, and the increasing cost of ensuring security diverts resources away

30. Richard D. Pethia, "Cyber Security—Growing Risk from Growing Vulnerability" 25 June 2003, Testimony before the House Select Committee on Homeland Security, Subcommittee on Cybersecurity, Science and Research and Development—hearing on "Overview of the Cyber Problem—A Nation Dependent and Dealing with Risk", <http://www.cert.org/congressional_testimony/Pethia_testimony_06-25-03.html> [Pethia, "Cyber Security"].

31. *Ibid.*

32. *Ibid.*

33. United States Nuclear Regulatory Commission, Office of Nuclear Reactor Regulation, Information Notice 2003-14, "Potential Vulnerability of Plant Computer Network to Worm Infection" (29 August 2003).

34. Kevin Poulsen, "US Warns Nuke Plants of Worm Threat" *SecurityFocus* (3 September 2003), <<http://www.securityfocus.com/news/6868>>.

35. Information Notice 2003-14, *supra* note 33.

from other important objectives, including maintaining and expanding networks.³⁶

DDOS attacks specifically, and cyber attacks generally, impose a range of less obvious social costs. For example, online service providers known as "blocklists," which filter out spam for their customers, have recently come under DDOS attacks believed to be intended to force them off the internet.³⁷ Blocklist service providers have been forced to discontinue their services or to purchase additional bandwidth to deal with the high level of attack traffic.³⁸ In addition, DDOS attacks are sometimes used as a means of private censorship to silence speakers whose opinions are abhorrent to the attackers. For example, during the recent Iraq war, the English-language web site of the Qatar-based satellite news service, "al Jazeera," was disabled by a DDOS attack.³⁹ Without taking a final position on the net social value or cost of "hactivism," attacks that disable sources of information considered objectionable impose costs on the "marketplace of ideas." The loss of unpopular voices may unduly simplify the debate on complex and important questions, thus impoverishing democratic political debate.

*

3. PREVENTION OF DDOS ATTACKS

3.1. Introduction

A DDOS attack involves many parties. These include: (1) the software vendor who rushes insecure software to market, (2) the user who does not apply patches to the software as vulnerabilities are discovered and patches announced and who does not update his or her virus scanner or use a firewall, (3) the ISPs who fail to scan email attachments for malicious code or block ports on customers' computers, (4) the person who takes advantage of the accumulated vulnerabilities in the system by writing and propagating malicious code, and eventually by launching a DDOS attack (the "mastermind"), and (5) the victim of the DDOS attack. Each of these parties could take steps with varying effectiveness to reduce the likelihood of DDOS attacks. This paper will work backwards through this chain of parties, considering the efficacy of any preventative measures available to each party. For reasons discussed in more detail in the final subsection, which deals with software developers, this paper takes the position that it is likely most efficient to address the problem of DDOS attacks at its root causes, particularly software insecurity. Improvements in software security would not only help to reduce DDOS attacks, but would also reduce the incidence of other cyber-scourges, such as epidemics of malicious code.

36. Dan Verton, "UN Hosts Global InfoSec Forum" *Computerworld* (12 September 2003), <<http://www.computerworld.com/2003/0,4814,84846,00.html>>.

37. Hiawatha Bray, "Saboteurs Hit Spam's Blockers" *The Boston Globe* (28 August 2003), <http://www.boston.com/news/nation/articles/2003/08/28/saboteurs_hit_spams_blockers/>.

38. *Ibid.*

39. Paul Roberts, "Al Jazeera Hobbled by DDOS Attack" *Infoworld* (26 March 2003), <http://www.infoworld.com/article/03/03/26/HNjazeera_1.html>.

3.2. *The Victim of a DDOS Attack*

There is essentially nothing that one can do to avoid becoming a victim of a DDOS attack.⁴⁰ The preventative options available to the target of a DDOS attack are largely unsatisfactory.⁴¹ It may choose to absorb the attack by maintaining large excess capacity that can handle the increased traffic of a DDOS attack. It can design its network so that capacity can be shifted to critical services while other services are degraded or disabled during the attack. It can also shut down all services until the attack has subsided. Attempts to filter out attacking traffic may also block legitimate traffic at the same time.⁴² As Householder notes,

[e]ven security-conscious sites can suffer a denial of service because attackers can control other, more vulnerable computer systems and use them against the more secure site. Thus, although you may be able to “harden” your own systems to help prevent having them used as part of a distributed attack, currently available technology does not enable you to avoid becoming a victim. There is some hope for the future in technological and other approaches.⁴³

As a result, one cannot look to the victims of DDOS attacks to protect themselves.

3.3. *The “Masterminds”*

Working backward through the other parties listed in the introduction to this Part, attempts could be made to reduce DDOS attacks by pursuing the masterminds who prepare the ground by infecting vulnerable computers to create “zombies,” and who use them to launch the DDOS attacks. Thus far, however, this has proven to be very difficult because of the inadequacies of tracing methods. The following observations note that these inadequacies reduce the deterrent effect of computer crime laws.

Although promising, research on tracking and tracing cyber-attacks is in a nascent state. The lack of proven techniques for effectively and consistently tracking sophisticated cyber-attacks to their source (and rarely to the individuals or entities responsible) severely diminishes any deterrent effect. Perpetrators feel free to act with nearly total anonymity.⁴⁴

40. Pethia, *supra* note 16. Pethia stated that “[a]lthough an organization may be able to ‘harden’ its own systems to help prevent having its systems used as part of a distributed attack, there is essentially nothing a site can do with currently available technology to prevent becoming a victim of, for example, a coordinated network flood.”

41. Householder, *supra* note 9 at 5.

42. *Ibid.* at 13:

It is often impossible for your [Internet service] provider to trace the attack beyond a certain ingress point in their networks, resulting in filters being applied on your behalf that also filter legitimate traffic. This can impact the subset of your customers whose traffic also reaches you through the filtered networks. Thus there are tradeoffs that must often be considered in order to ride out an attack with as much grace as possible.

43. *Ibid.* at 20.

44. Howard F. Lipson, “Tracking and Tracing Cyber-Attacks: Technical Challenges and Global Policy Issues” CERT Coordination Center, Special Report CMU/SEI-2002-SR-009, (Carnegie Mellon University) at 49.

It is easy for attackers to avoid getting caught by hiding their identity. They command their attack network from stolen dial-up accounts and other compromised systems, and they use spoofed source addresses for attack traffic. Victim sites and law enforcement face a daunting and frequently unfeasible task to identify and prosecute attackers. Suffering few consequences—if any—for their actions, attackers continue their work.⁴⁵

The pursuit of the masterminds is further complicated by the fact that they may be launching their attacks from foreign jurisdictions.

Even if one does try to pursue the masterminds, this should not be the sole approach to preventing DDOS attacks. An exclusive focus on deterring masterminds places the full burden of preventing DDOS attacks on law enforcement agencies, and provides inadequate incentives for others, such as software vendors or users, to exercise care to reduce vulnerabilities.

Another way to impede the actions of masterminds would be to suppress or, at least, to delay the public release of information about software vulnerabilities. There is a lively debate over the propriety of finding and disclosing software vulnerabilities.⁴⁶ Those doing so argue that this is justified for several reasons. First, cyber-criminals might otherwise find and exploit the flaws secretly. Public disclosure puts the public on notice and allows them to take preventive steps. Second, software vendors are slow to prepare patches for vulnerabilities if the existence of the vulnerability is communicated quietly to the vendor. Public exposure is thought to be necessary in order to provoke a speedy reaction by software vendors.

A group comprised of eleven security companies and software developers, known as the Organization for Internet Safety, recently released a consensus document setting out guidelines for reporting security vulnerabilities.⁴⁷ The guidelines suggest that software vendors be given a thirty-day grace period after being notified of the existence of a vulnerability before there is public disclosure of that vulnerability. This delay is meant to permit the development and installation of patches. Other aspects of this approach have been criticized. The guidelines specify that the software vendor should restrict the release of the technical details of the vulnerability to a short list of businesses such as antivirus vendors while a patch is being developed.⁴⁸ While this makes some sense, it has been observed that this practice “benefits some companies by increasing the value of paid security mailing lists, while harming academic research into security problems.”⁴⁹

Another suggestion, aimed at making it more difficult for cyber-criminals to operate, is to pursue the web sites that offer information and tools used to create malicious code and to launch attacks. Some have suggested that on-line

45. Householder, *supra* note 9 at 23.

46. See e.g., Search Security, “User Comments on Full Disclosure of Software Vulnerabilities” *Search Security* (2 July 2002), <http://searchsecurity.techtarget.com/originalContent/0,289142,sid14_gci837110,00.html>. See also Robert Vamosi, “My plan for fixing software flaws” *CNET News.com* (2 October 2002), <http://reviews.cnet.com/4520-3513_7-5021280-1.html>.

47. Dan Verton, “Vendor Group Publishes Vulnerability Disclosure Guidelines” *Computerworld* (31 July 2003), <<http://www.computerworld.com/2003/0,4814,83609,00.html>>.

48. Andrew Brandt, “Security Flaws Under the Microscope” *Computerworld* (7 August 2003), <<http://www.computerworld.com/securitytopics/security/story/0,10801,83811,00.html>>.

49. *Ibid.*

sharing of such information should be criminalized.⁵⁰ Problems with this approach include the difficulty of distinguishing between malicious information and information that could be used for legitimate security research, as well as freedom of speech issues.⁵¹

3.4. Internet Service Providers

Continuing to work backward through the chain of parties involved in a DDOS attack, it has been suggested that ISPs could help to reduce the spread of the malicious code that is used to create the army of “zombie” computers. ISPs could be required to scan all email attachments for computer viruses before they are sent to subscribers. However, scanning is resource-intensive, and may expose ISPs to liability if they mistakenly filter out legitimate email.⁵² Others warn that placing responsibility for scanning on ISPs would create serious delays.⁵³ Notwithstanding these concerns, several major ISPs already scan subscriber email for viruses.⁵⁴

In any event, the development of worms, like the recent MSBlaster worm,⁵⁵ that spread through direct internet connections rather than by email attachment, means that email filtering would not be a sufficient answer. A recent report published by the SANS Institute suggests that ISPs should permanently block those ports on their customers’ computers that are commonly used for malicious traffic.⁵⁶ The MSBlaster worm used such ports to find and infect vulnerable computers running the Windows operating system.⁵⁷ Many ISPs already block some of these ports, while others offer free personal firewall software for customers to install.⁵⁸ It is perhaps sensible to block ports that many unsophisticated computer users do not need. However, this approach may contribute to the unfortunate degeneration of the computer from a powerful and flexible machine into an “appliance” used by a largely passive audience, as creative control and autonomy are increasingly removed from the computer user.⁵⁹

50. Kim Zetter, “Just Say No To Viruses and Worms” *InfoSec News* (11 September 2003), re-posted from <<http://www.wired.com/news/infostructure/0,1377,60391-2,00.html>>.

51. *Ibid.*

52. Brian Krebs, “Preventive Medicine for E-Mail” *The Washington Post* (28 August 2003) E04.

53. Cynthia L. Webb, “Sue Microsoft?” *The Washington Post* (12 September 2003), <http://www.washingtonpost.com/ac2/wp-dyn?pagename=article&node=&contentId=A64745-2003Sep12 ¬Found=true_washingtonpost.com>.

54. Krebs, *supra* note 52.

55. See CERT, “CERT@ Advisory CA-2003-20 W32/Blaster worm” last revised 14 August 2003, <<http://www.cert.org/advisories/CA-2003-20.html>>, for a description of this worm.

56. Johannes Ullrich, “Internet Service Providers: The Little Man’s Firewall?” SANS Institute, <http://www.sans.org/rr/special/isp_blocking.pdf>.

57. Paul Roberts, “Study: ISPs Should Block ‘Net Attack Ports,” *NetworkWorldFusion—IDG News Service* (8 September 2003), <<http://www.nwfusion.com/edge/news/2003/0908studyisps.html>>.

58. *Ibid.*

59. Professor Zittrain discusses the concern that the computer may turn into an inflexible “appliance” due to the pressure to improve security as well as to the pressure from the copyright industries to control access to their content. See Jonathan L. Zittrain, “Taming the Consumer’s Computer” *The New York Times* (11 March 2002), <<http://www.nytimes.com/2002/03/11/opinion/11ZITT.html?ex=1075698000&en=bf5eb2351ddf519b&ei=5070>>.

3.5. Internet-Connected Computer Users

There are a variety of steps that can be taken by the owners of internet-connected computers to avoid the takeover of their computer for use as launching pads for DDOS attacks. Regular patching of vulnerabilities, as well as regular scanning for infection with malicious software, will help to ensure that the computer does not become one of the "zombies" in a DDOS mastermind's army. The installation of firewalls that monitor outgoing traffic can catch packets leaving users with "spoofed" source addresses.⁶⁰ Since DDOS attacks often use these spoofed source addresses to conceal the source of an attack, and legitimate traffic will always bear a source address from within the address space assigned to the user, such "egress filtering" by the user can help to reduce outgoing DDOS attack packets.⁶¹

One of the key steps that computer users could take to prevent DDOS attacks is to patch their software. If vulnerabilities have been patched, masterminds will be unable to exploit them to recruit the computer for later use in a DDOS attack. Software vendors offer updates and patches via the internet to remedy problems in their software products as they come to light. Unfortunately, the frequency with which bugs and vulnerabilities come to light is very high. The CCIA suggests that Microsoft is issuing patches at the rate of one every six days.⁶² Furthermore, the window between the announcement of a software vulnerability and the release of a virus that exploits the vulnerability has been narrowing.⁶³ This means that there is a decreasing window within which to issue and apply patches to all vulnerable machines. As a result, the burden of patching faulty software is increasingly intolerable to users, large and small alike, who are falling behind in patching their software. It is not only the unsophisticated individual user who finds it difficult to keep up.

Not every IT department keeps up to speed on the vulnerabilities, nor do they always patch immediately. Witness the large number of organizations that are hit by viruses exploiting long known vulnerabilities. We are suffering from patch overload. If you implemented all patches as received would there be any time left to actually run the computer systems and develop them for the organization?⁶⁴

In addition to the burden caused by the frequency of patches, there are problems with both patch quality and the patching process. The patches themselves may be incompatible with existing software or configurations, or they may

60. Householder, *supra* note 9 at 18.

61. *Ibid.*

62. Computer & Communications Industry Association (CCIA), "CyberInsecurity: The Cost of Monopoly" (24 September 2003) at 15 <<http://www.cciainet.org/papers/cyberinsecurity.pdf>>. See also Todd Bishop "Should Microsoft Be Liable For Bugs?" *Seattle Post-Intelligencer* (12 September 2003), <http://seattlepi.nwsource.com/business/139286_msftliability12.html>, suggesting a similar rate (thirty-nine in approximately the first eight and a half months of 2003).

63. Zetter, *supra* note 50. See also Linda McCarthy, ed., Symantec Internet Security Threat Report, Executive Summary, Symantec Corporation (September 2003), <<http://enterprisesecurity.symantec.com/content.cfm?ArticleID=1539>>.

64. Stephen Hinde, "Compsec 2002: The Complete Security Circle" (2002) 21 *Computers & Security* 689 at 693.

not successfully fix the vulnerability.⁶⁵ In fact,

[s]tories abound about service packs or patches causing problems with computers, which in turn cause businesses to be leery about applying them... At the same time, most enterprises do not have the capability to do a risk assessment and determine whether they do indeed need a patch for a particular vulnerability.⁶⁶

Furthermore, some users complain that patches are too big for convenient download when using dialup connections.⁶⁷

Even if computer users are patching their systems, there is increasing concern that hackers will target the patch download sites and insert malicious code into the patch, or redirect users to a false software vendor download site offering malicious software masquerading as a patch.⁶⁸ In fact, there have been a number of attacks against open-source software distribution sites in 2002 and into 2003.⁶⁹ There have also been reports that viruses have been sent in email messages that purport to be official notifications of new Microsoft patches.⁷⁰

Another serious concern with the patching and updating system is that it may be used to force unwelcome changes in the software licence. In 2002, the Register reported that Microsoft had included a licence change within updates to Windows XP and Windows 2000.⁷¹ The new licence condition required users to consent to Microsoft's automatic scanning and updating of the operating system and/or its components. The condition states:

You acknowledge and agree that Microsoft may automatically check the version of the OS Product and/or its components that you are utilizing and may provide upgrades or fixes to the OS Product that will be automatically downloaded to your computer...⁷²

-
65. Patrick Gray, "Security Firm: IE Patch Does Not Work" *CNET News.com* (8 September 2003), <<http://news.com.com/2100-1009-5072672.html>>. Patrick Gray reports that a patch was issued by Microsoft on 20 August 2003 for a flaw reported the previous spring. The patch caused problems for non-default operating system configurations, and so it was re-released on 28 August 2003. Security experts at eEye Digital Security reported that the patch does not fix the vulnerability. See also Robert Lemos, "Microsoft Fixes Broken Patch" *CNET News.com* (6 October 2003), <http://news.com.com/2100-1002_3-5086979.html>.
66. Jay Wrolstad, "Industrious Worm vs. Lazy IT?" *NewsFactor Network* (10 September 2003), <<http://www.newsfactor.com/perl/story/22253.html>>, quoting Russ Cooper of TruSecure.
67. Michelle Delio, "Worm Exploits Weak Link: PC Users" *Wired News* (13 August 2003), <<http://www.wired.com/news/infrastructure/0,1377,59994,00.html>>.
68. Elias Levy, "Poisoning the Software Supply Chain" *IEEE Security & Privacy* (May/June 2003) <<http://ieeexplore.ieee.org/iel5/8013/27102/01203227.pdf?isNumber=27102&prod=JNL&arnumber=1203227&arSt=+70&ared=+73&arAuthor=Levy%2C+E.>> 70. False web sites have been in vogue in 2003. In these scams, bank customers have been sent email directing them to a fake site designed to resemble the legitimate bank web site. They are then asked to enter personal financial details. See David Legard, "Fake Bank Web Site Scam Reaches U.S." *Computerworld* (14 May 2003), <<http://www.computerworld.com/2003/0,4814,81211,00.html>>.
69. Levy, *ibid.*
70. Delio, *supra* note 67; Robert Vamosi, "Sven Prevention and Cure: Sven Virus Masquerades as a New Microsoft Patch" *CNET Reviews* (18 September 2003), <http://reviews.cnet.com/4520-6600_7-5078675.html>.
71. Andrew Orlowski, "Microsoft EULA Asks For Root Rights—Again" *The Register* (2 August 2002), <<http://www.theregister.co.uk/content/4/26517.html>>.
72. *Ibid.*

The Register reported on a previous occurrence as well, involving a patch for security vulnerabilities in Windows Media Player.⁷³ In that case, the licence modification required users to consent to the automatic installation of digital rights management controls on their systems. That license stated:

You agree that in order to protect the integrity of content and software protected by digital rights management ('Secure Content'), Microsoft may provide security related updates to the OS Components that will be automatically downloaded onto your computer. These security related updates may disable your ability to copy and/or play Secure Content and use other software on your computer. If we provide such a security update, we will use reasonable efforts to post notices on a web site explaining the update.⁷⁴

Although users may refuse these licence modifications (assuming they notice them), they would then be unable to obtain necessary updates and patches.⁷⁵

Finally, there is evidence that patching is sometimes inadvertently undone when systems are reinstalled from backup copies. Gerhard Eschelbeck, of security software company Qualys Inc., released an interesting study of security vulnerabilities at the Black Hat Briefings in Las Vegas in the summer of 2003.⁷⁶ Eschelbeck monitored his customers' networks for 18 months for common security problems. He found that a security vulnerability typically declines sharply following an epidemic of a virus that targets that vulnerability, but that it gradually re-emerges when attention to the virus ebbs away. He suggests that information technology "IT" departments, which keep backup images of hard drives to facilitate restoration of a laptop or desktop to company default settings, probably fail to update these images with patches. As a result, when a computer has an old image reinstalled, the old vulnerabilities are reinstalled as well.

In any event, if businesses with IT staff are not always reliably patching their software, unsophisticated home-users are even less well equipped to do so.⁷⁷ There have been several suggestions aimed at resolving this problem. One suggestion is that users' computers be *automatically* patched by software vendors via the internet. Another suggestion is that users who fail to patch their soft-

73. Thomas C. Greene, "MS Security Patch EULA Gives Billg Admin Privileges on Your Box" *The Register* (30 June 2002), <<http://www.theregister.co.uk/content/4/25956.html>>.

74. *Ibid.*

75. Richard Forno, "Overcoming 'Security by Good Intentions'" *The Register* (6 September 2003), <<http://www.theregister.co.uk/content/55/31094.html>>.

76. Brandt, *supra* note 48.

77. Pethia, "Cyber Security" *supra* note 30 under *Vulnerability of the Internet and World Wide Web*. Pethia stated that patches and upgrades often are not applied either because "the job may be too time-consuming, too complex, or just at too low a priority for the system administration staff to handle." See also Bruce Schneier, "Fixing Network Security by Hacking the Business Climate" (June 2002), <<http://www.counterpane.com/presentation4.pdf>> at 27. As for the unsophisticated end-user, Schneier notes that "there are just too damn many patches. It's simply impossible to keep up...I don't know how the average user can possibly install them all; he'd never get anything else done."

were be punished by a fine or the disablement of their internet connections.⁷⁸ Stephen Henderson and Matthew Yarbrough have suggested that ISPs could be required to become involved in the update process by hosting updates and scanning subscribers' systems for insecurities.⁷⁹ They have also suggested that those users whose computers are used in attacks should be held liable in negligence if they have failed to meet reasonable security standards.⁸⁰

Due to the general failure of many users to patch vulnerabilities, Microsoft is looking at setting future Windows programs to accept software patches *automatically* without user approval, unless a user specifically opts out.⁸¹ It is likely that the Auto Update function will be imposed on home users but not on businesses, which may be more resistant due to the fear that the patches could interfere with existing programs.⁸² Many notable internet security experts support this approach, noting that "...it gets the updates out to the non-technically savvy masses, and that's the majority of Internet users," and that "[s]ecurity is a trade-off, to be sure, but this is one trade off that's worthwhile."⁸³ Critics are concerned, however, that automatic updates would enable Microsoft surreptitiously to make changes to the operating system for other reasons, such as to block access to certain content.⁸⁴

Of course all of the concerns regarding patch quality, patch security and the misuse of the patching process to force changes to license agreements exist, perhaps to a heightened degree, with an automatic patch system. Although it would not resolve the other concerns, one solution to the misuse of the patching process to impose license changes or to make unwanted changes to the software would be to establish a trusted process whereby patches could be vetted before being put into circulation.

There are several problems with measures, such as lawsuits or penalties, that target the computer user directly. If measures were aimed at the owners of computers used in a DDOS attack, it would be necessary to deal with the problem of tracing the responsible computers, given the prevalence of spoofed addresses. If measures were aimed at the owners of infected or vulnerable computers, one must confront the fact that the owners of infected computers may be located beyond the jurisdictional reach of the authority seeking to punish or sue them. It would also be expensive to pursue the large numbers of owners of infected computers, although a small number of lawsuits or fines might have a

78. Universities have employed a variety of methods to encourage students to maintain the security of their computers. Some disable the accounts of infected users and fine those who inadvertently spread viruses. Others impose virus scrubbing fees or require signed statements that security patches are up to date. See Associated Press, "Colleges Crack Down on Viruses," *Wired News* (4 September 2003), <<http://www.wired.com/news/technology/0,1282,60299,00.html>>. See also Brian Krebs "Universities Rush to Protect Networks" *The Washington Post* (4 September 2003), <<http://www.washingtonpost.com/ac2/wp-dyn?pagename=article&node=&contentId=A25845-2003Sep4¬Found=true>>.

79. Stephen E. Henderson & Matthew E. Yarbrough, "Frontiers of Law: The Internet and Cyberspace: Suing the Insecure?: A Duty of Care in Cyberspace" (2002) 32 N.M.L. Rev. 11 at 23.

80. *Ibid.*

81. Brian Krebs, "Microsoft Weighs Automatic Security Updates as a Default" *The Washington Post* (19 August 2003), <<http://www.washingtonpost.com/ac2/wp-dyn/A11579-2003Aug18>>.

82. *Ibid.*

83. *Ibid.* quoting Bruce Schneier.

84. *Ibid.*

broad deterrent effect on users generally. Nevertheless, it may sometimes be effective to pursue the owners of “zombie” computers, particularly where a significant portion of the attack can be traced to one group of “zombies” such as computers at a university or corporation.⁸⁵

3.6. Software Developers

Finally, the last link in the chain of parties implicated in a DDOS attack is the developer of insecure software. A recurring theme in discussions of cyber security is that poor software design is one key cause of the cyber security problem. The inadequacy of much software is said to flow from an excessive focus within the software industry on designing complex features⁸⁶ and speed to market,⁸⁷ rather than on security.

Software insecurity is an intuitively appealing issue to tackle. It seems more sensible to fix a problem at its source (namely, the security vulnerabilities in the underlying software) than to try to apply preventative and defensive measures once the software has been widely deployed in the market.⁸⁸ “In simple economic terms, finding and removing bugs in a software system before its release is orders of magnitude cheaper and more effective than trying to fix systems after release.”⁸⁹

Which is easier? To get a couple thousand professional paid software engineers with at least four years of training each to follow established guidelines for writing secure code, or to get many millions of users with a wide variety of levels of technical expertise to make informed decisions about whether or not to exercise one of the functions for which the software they paid money to acquire is designed?⁹⁰

The expectation that it would be cheaper to tackle security by designing secure software seems intuitively correct. However, this expectation should be examined more closely. For example, it may be more efficient for a highly competitive and specialized group of security researchers to compete to locate and protect against vulnerabilities in software than for one software vendor to

85. Henderson & Yarbrough, *supra* note 79 at 14 that “Mafiaboy’s [the attacker implicated in the February 2000 DDOS attacks] target of choice for zombies was universities, which are often insecure and have a significant amount of computing power sitting idle at any given time.”

86. Schneier, *supra* note 77 at 7:

The technology industry is driven by demand for features, for options, for speed. There are no standards for quality or security, and there is no liability for insecure software. Hence, there is no economic incentive to create high quality. Instead, there is an economic incentive to create the lowest quality the market will bear. Unless customers demand higher quality and better security, this won’t change.

87. Householder, *supra* note 9 at 23, writes that “[c]urrent economic pressures lead vendors to focus on achieving a fast time to market rather than on designing secure networks and applications. Without some financial (or legal) incentive to behave more securely, developers will continue to produce vulnerable products.”

88. G. Gary McGraw, “On Bricks and Walls: Why Building Secure Software is Hard” (2002) 21:3 *Computers & Security* 229. McGraw writes that “[c]urrent approaches, based on fixing things only after they have been exploited in fielded systems, address only symptoms, ignoring the cause of the problem.”

89. *Ibid.* at 231, referring to F. Brooks Jr., *The Mythical Man-Month: Essays on Software Engineering*, 2d ed. (New York: Addison-Wesley, 1995).

90. Michelle Delio, “Geeks Grapple With Virus Invasion” *Wired News* (21 August 2003), <<http://www.wired.com/news/infostructure/0,1377,60109,00.html>>.

look for and remedy vulnerabilities in its software products. On the other hand, even if more efficient testing may be achieved by harnessing the competitive energies of cyber security researchers, it may still be better for the software vendor to adopt secure software design procedures and to conduct thorough security testing itself before releasing the software. This is because cyber security companies would be interested in testing software only *after* it has been widely deployed. They can then be assured of the largest market or the greatest publicity for their efforts. Any flaws thus discovered would have to be patched using the inadequate and costly patching systems. As a result, any testing efficiencies flowing from leaving testing to a secondary cyber security market may be obliterated by the greater costs of remedying the problems discovered. Furthermore, to the extent that the secondary cyber security market tests for software security flaws as well as sells products such as virus scanning software, their interests may lie, not in improving software security, but in supplying defences to the exploitation of software vulnerabilities (such as virus scanning software).⁹¹ If software design were improved, some of the social resources currently devoted to the development of products designed to protect against the exploitation of the software flaws would be saved. The cyber security market also addresses insecurities not attributable to software flaws and so the total social resources consumed in pursuing cyber security would not all be saved if software were better designed. Nevertheless, there could be some savings, and it is therefore worth noting the considerable amount of resources devoted to cyber security today. Gartner estimates that the global IT security market for hardware, software and services is now worth \$17 billion, and is set to grow at an annual rate of 15% for the foreseeable future.⁹²

Even if it is currently cheaper to deal with cyber insecurity by applying *ex post facto* patches and defensive measures than it is to design more robust software in the first place, this balance may reverse in the future. Computer security expert Bruce Schneier suggests that the traditional approach to computer security, which consists of the application of technological band-aids, is inadequate given the worsening state of computer security.⁹³

This article will therefore proceed on the assumption that it is preferable to address the problem of cyber insecurity at the level of software design and development. As noted earlier, various cyber security experts suggest that software flaws are one of several important focus points in any effort to improve internet security.

Software vendors seem so far to have done a rather poor job of learning from past experience with software vulnerabilities. The Director of CERT, Richard Pethia, recently noted that “[t]here is little evidence of improvement in the security features of most products...” and “...developers are not devoting

91. See e.g. Joanna Glasner, “Security Biz Thrives on Fear” *Wired News* (16 April 2003), <<http://www.wired.com/news/infostructure/0,1377,58492,00.html>>.

92. Alex Salkever, “Microsoft, Your PC’s Security Guard?” *Business Week online* (14 August 2003), <http://www.businessweek.com/technology/content/aug2003/tc20030814_6379_tc047.htm>.

93. Schneier, *supra* note 26, slides 9–10.

sufficient effort to apply lessons learned about the sources of vulnerabilities."⁹⁴ Pethia further stated that, "[w]e continue to see the same types of vulnerabilities in newer versions of products that we saw in earlier versions. Technology evolves so rapidly that vendors concentrate on time to market, often minimizing that time by placing a low priority on security features."⁹⁵

Pethia also described the rapid increase in the reporting of new vulnerabilities to CERT between 1995 (140 new reports annually) and 2002 (a little over 4,000 new reports).⁹⁶ Another 2,982 vulnerabilities were reported in 2003.⁹⁷

Pethia suggested that one important way in which the state of cyber security can be improved is for software developers to follow good software engineering principles.⁹⁸ The Computer Science and Telecommunications Board ("CSTB"), a division of the U.S. National Research Council, stresses that software vendors should "[s]trengthen software development processes and conduct more rigorous testing of software and systems for security flaws, doing so before releasing products rather than use customers as implicit beta testers to shake out security flaws."⁹⁹

The costs of software security vulnerabilities are high. This paper discussed the costs of DDOS attacks in Part 2. Yet DDOS attacks are not the only costs associated with insecure software. The costs of malicious code, some of which capitalizes on software flaws, are also high. The London-based computer security firm mi2g Ltd. estimates global damage from malicious software in 2003 will amount to \$107 billion. The same firm estimates that the SoBig worm attack in August 2003 cost \$30 billion on its own.¹⁰⁰ Another advantage, therefore, in addressing DDOS attacks at their root cause (namely, software flaws) is that the additional costs of malicious code in general will be reduced at the same time.

However, it is important to note that increased software security will also generate additional costs. Secure design standards and testing would likely raise the cost of software design, could delay the development time for new programs and might constrain the inclusion of innovative features.¹⁰¹ In other words, greater security may result in higher prices as well as in reductions in certain other aspects of quality.

In a properly functioning competitive market, one would expect software to be designed with the optimal balance of price and quality (including security characteristics). The widespread view that software is excessively insecure suggests that the market, as it presently operates, may not be generating this optimal balance. There are a number of possible market imperfections to consider. First, the markets for certain key pieces of software may not be per-

94. Pethia, *supra* note 30.

95. *Ibid.*

96. *Ibid.*

97. CERT/CC, *supra* note 2.

98. Pethia, *supra* note 30.

99. CSTB, *supra* note 8 at 14.

100. CCIA, *supra* note 62 at 10, referring to David Zeiler, "Government Issue" *Baltimore Sun/SunSpot.net* (18 September 2003).

101. CSTB, *supra* note 8 at 10 that "[t]here are often tensions between security and other good things, such as features, ease of use, and interoperability."

fectly competitive. Second, purchasers are generally incapable of assessing the price/quality balance of extremely complicated products such as modern software. Although these “information costs” might otherwise be met by secondary information providers (such as publishers of product comparisons and reviews), this solution is sometimes suppressed by “anti-benchmarking” clauses in software licences. Third, a purchaser of software will not bear all of the costs of insecure software, which are instead inflicted on other internet-connected third parties. This “insecurity externality” reduces the amount that a purchaser will be willing to invest in security to a point that is below the social optimum.¹⁰² Each of these possible problems will now be discussed at greater length.

It is clear that certain key software programs such as Windows¹⁰³ and Internet Explorer¹⁰⁴ have achieved near complete market share. There may be many reasons for this state of affairs other than the attractiveness of the product, such as the existence of strong economies of scale, network externalities (particularly where interoperability with competitors’ programs is impeded by the dominant incumbent), as well as high switching costs.¹⁰⁵ There is, however, some debate about whether these characteristics have led to market failure in the software context.¹⁰⁶ In any event, the ubiquity of these programs among networked computers raises a most interesting problem, namely the vulnerability of homogeneous systems. The biological sciences have revealed that genetically homogeneous populations are highly vulnerable to pests and disease, for the understandable reason that members of the homogeneous population all share the same weaknesses. This offers a large evolutionary reward to organisms that evolve to take advantage of the weaknesses, while, at the same time, ensuring that the whole population could be harmed should such an opportunistic organism evolve. This analogy to biology works very well in the realm of cyber security. The Computer & Communications Industry Association¹⁰⁷ (“CCIA”) released a report in September 2003 entitled “*CyberInsecurity: The Cost of Monopoly*,” which warns that the near complete reliance on the Microsoft operating system makes most of the world’s computers vulnerable to the same worms and viruses at the same time. The CCIA suggests that,

102. An externality is a cost or benefit conferred on a third party by an activity of producers or consumers in the market. See David Gowland & Anne Paterson, *Microeconomic Analysis: A Modern Introduction* (New York: Harvester Wheatsheaf, 1993) at 287.

103. OneStat.com reported on 24 September 2003 that Microsoft’s Windows operating system has 97.34% of the global market, while Apple Macintosh has 1.49% and Linux has 0.51%. <http://www.onestat.com/html/aboutus_pressbox24.html>.

104. OneStat.com reported on 28 July 2003 that Microsoft’s Internet Explorer browser has 95.4% of the global market, while Netscape Navigator has 2.5% and Mozilla has 1.6%. <http://www.onestat.com/html/aboutus_pressbox23.html>.

105. Klaus M. Schmidt & Monika Schnitzer, “Public Subsidies for Open Source? Some Economic Policy Issues of the Software Market” (2003) 16 Harv. J.L. & Tech. 473 <<http://jolt.law.harvard.edu/articles/pdf/v16/16HarvJLTech473.pdf>> .

106. See David S. Evans & Bernard J. Reddy, “Government Preferences for Promoting Open Source Software: A Solution in Search of a Problem” (2003) 9 Mich. Telecomm. Tech. L. Rev. 313.

107. The CCIA’s criticisms of Microsoft have been dismissed by a rival group, the Association for Competitive Technology (of which Microsoft is a member), as motivated not by concern for security but by the desire to undermine Microsoft. See Todd R. Weiss, “Rival Groups Debate DHS Deal With Microsoft” *Computerworld* (28 August 2003), <<http://www.computerworld.com/2003/0,4814,84434,00.html>>.

[b]ecause Microsoft's near-monopoly status itself magnifies security risk, it is essential that society become less dependent on a single operating system from a single vendor if our critical infrastructure is not to be disrupted in a single blow. The goal must be to break the monoculture.¹⁰⁸

Steve Ballmer, CEO of Microsoft, has noted that the "Windows (operating system) is the most popular platform in the world, so every security incident with it is just magnified and magnified and magnified across so many more systems than with any other platform."¹⁰⁹

This observation may have been meant to suggest that security flaws in a widespread software program will receive greater publicity than flaws in a less widely-used program, perhaps leading to the inaccurate perception that the dominant program is less secure than any competing programs. However, this does not contradict the observation that spectacular and widespread failures are more likely within a monoculture, whether of software or biological organisms.

The CCIA argues that competition policy must take into consideration the security implications of a software monopoly.¹¹⁰ If the monopoly is inevitable, the CCIA argues, the monopolist should be required to provide compensation for security-related flaws.

If governments do not dismantle the monopoly but choose instead to modify the practices of the monopoly they must concede that that route will, like freedom, require eternal vigilance. Appropriate support for addressing the security-related pathologies of monopoly would doubtless include the introduction of effective, accessible rights of action in a court of law wherever security flaws lead to harm to the end-user. In extreme cases, the consequences of poor security may be broad, diffuse, and directly constitute an imposition of costs on the user community due to the unfitness of the product. Under these circumstances, such failures should surely be deemed "per se" offenses upon their first appearance on the network.¹¹¹

The analogy between genetic and software code, between biological pathogens and digital pathogens, and so on, may become even more apt with the development of evolutionary computing techniques.¹¹² The problem of cyber security thus suggests another reason for competition law to be concerned with monopolies in addition to the more usual concern that monopolists will raise prices and restrict output to below the efficient level.

Software markets, particularly those for mass-marketed software, also suffer from information failures. A perfectly functioning market assumes that market decisions are made under conditions of perfect information about the

108. CCIA, *supra* note 62 at 5.

109. "Ballmer: Microsoft 'Humbled' By Security Woes" *Associated Press* (16 September 2003), <<http://www.crn.com/sections/BreakingNews/dailyarchives.asp?ArticleID=44542>>.

110. CCIA, *supra* note 62 at 18.

111. *Ibid.*

112. Evolutionary computing is a problem-solving method that uses algorithms modeled on natural genetic and evolutionary mechanisms. See e.g. William M. Spears *et al.*, "An Overview of Evolutionary Computation" 1993 European Conference on Machine Learning, <<http://www.cs.uwoy.edu/~wspears/overview/ecml93.all.html>>.

price and quality of products.¹¹³ This assumption rarely holds. To the extent that the consequences of an incorrect market decision justify the cost of obtaining information, a purchaser will either inform him or herself or take advantage of a secondary market for information (e.g. publications containing product comparisons and reviews). In the case of software, the average end-user is capable of assessing only the least sophisticated aspects of software quality, and instead relies on expert intermediaries for advice on the relative merits of competing programs. The practice of some software vendors to include “anti-benchmarking” provisions within the license agreements governing the use of the software weakens the capacity of the secondary information market to educate the actual purchaser.¹¹⁴

In the context of DDOS attacks, a computer user’s poor computer security imposes negative externalities on others.¹¹⁵ A negative externality exists when a person’s production or consumption decision imposes costs on a third party for which the person does not have to pay.¹¹⁶ The market’s ability to generate the socially-optimal outcome is undermined when externalities exist. Where a person does not face the full costs of his or her choice, the person does not take those costs into consideration in making the choice and is likely to make that choice more than would be optimal from the perspective of society as a whole.¹¹⁷

In the case of DDOS attacks, there is little that a victim can do to avoid the attack. The attack is launched from an army of conscripted computers, whose under-investment in security made them vulnerable to conscription. The owners of the “zombie” computers are unlikely to face the risk of a DDOS attack. Even if the owners of the conscripted computers *did* face the risk of a DDOS attack, *they*, in turn, would rely on others to protect them from the attack and so the risk of a DDOS attack provides no incentive for them to invest in security. In other words, the threat of a DDOS attack offers no incentives to invest in security.

[U]nder today’s practices, a party that makes investments to prevent its own facilities from being used as part of a distributed denial-of-service (DDOS) attack will reap essentially no benefits from such investments, because such an attack is most likely to be launched against a different party. But today’s Internet-using society would clearly benefit if many firms made such investments. Making parties liable for not securing their facilities against being illicitly used as part of a DDOS attack (today there is zero liability) would change the incentives for making such investments.¹¹⁸

113. Gowland & Paterson, *supra* note 102 at 274–275.

114. For a recent example of this problem see *People v. Network Associates, Inc. dba McAfee Software* 195 Misc.2d 384; 758 N.Y.S.2d 466 (N.Y. Sup. Ct., 2003). The Court considered a clause within the end-user license agreement for certain security software programs that purported to bar users from publishing “product reviews” or disclosing the results of “benchmark tests” to any third party without the company’s prior consent. The dispute arose because an online magazine, Network World Fusion, published the results of a comparative test of firewall programs, in which Network Associates’ program did poorly. Network Associates apparently objected to publication and cited the license agreement.

115. Whether or not security investments have externality effects will depend on the type of cyber security problem being considered. For example, when one considers the theft of proprietary information from an insecure network, it is the owner of the information that will suffer the full consequences of failing to secure its networks.

116. Gowland & Paterson, *supra* note 102.

117. The Coase theorem suggests that the socially-optimal outcome might still be achieved if the actor and the third party affected by the action are able to bargain easily. In the present context, where there are enormous numbers of computer users, and a few potential target web sites, the transaction costs of negotiating optimal investment in security are very high.

118. CSTB, *supra* note 8 at 16, n. 13.

The threat of malicious software that harms a user's computer (rather than merely permitting the user's computer to be used to harm another's computer) might offer some incentive to invest in security, depending on whether the probability-discounted losses from virus infection outweigh the costs of security measures. Some will find it worthwhile and will take steps to protect their computers from malicious code. However, if the costs of DDOS attacks were internalized by computer owners, then more users, particularly those whose own personal calculation of the costs and benefits of self-protection place them near the borderline in deciding whether to invest in computer security, would invest in security. As a result, the failure to internalize the costs of DDOS attacks will likely mean a sub-optimal investment in security in the network as a whole, regardless of whether some users will invest in security out of self-protection impulses anyway.

In sum, there is a widespread sense that poor software design and development (particularly that of mass-marketed software relevant to internet use) is one key contributor to the insecurity of the internet. The inadequacy of the software may reflect a variety of market failures, rather than the socially optimal balance of price and quality.

In some situations, regulations or laws can help to fix market failures. For example, a government could require that particular software design standards be respected, effectively setting the level of security to be "bought" by all computer users. This approach has many disadvantages. It is likely to be far too blunt, given that different types of programs may present very different levels of risk, and so should not all be required to meet the same design standards. Furthermore, the regulatory process is slow and prone to "capture" by regulated industries.

Another approach would be to impose liability for negligently-designed software. Although private legal disputes are extremely costly, a negligence-based approach offers certain advantages. In particular, the standard of care required of software developers can be varied according to the context. Software intended for use in conditions where design flaws may lead to substantial losses may be treated differently from software that does not present high risks. In this way, the potential negative consequences of greater security, namely increased cost, slower development, reduced features and slower innovation, may be averted for less "mission-critical" software.

Nonetheless, there are some dangers in applying negligence liability for security defects in software. It is possible that the software industry could take steps that would be very effective in improving security, but that may have undesirable consequences of another sort. The idea of "trusted computing platforms" re-emerged in 2002 when chip makers and Microsoft began to explore the idea of embedding within computers a tamper-resistant security chip, which would then store information about authorized software.¹¹⁹ The operating system could then verify the trustworthiness of the software before permitting the software to be run.¹²⁰

119. Richardson, *supra* note 22 at 13.

120. *Ibid.*

Professor Zittrain points out the danger inherent in this approach.¹²¹ It uses “digital gatekeepers that act like bouncers outside a nightclub, ensuring that only software that looks or behaves a certain way is allowed in.”¹²² This offers the computer manufacturer or the operating system maker an opportunity to limit the kinds of software a computer can run. Professor Zittrain warns that the twin pressures of the search for security and reliability and the content industry’s desire to control what computer users can do with information will push the computer to become an inflexible, if dependable, “appliance.”

The PC platform and the Internet to which it connects is the engine of the information revolution—as important to our economy and culture as all the movies of Hollywood. A shift from open platforms to closed appliances may be inevitable, as our consumerist desire for trustworthy PCs dovetails with information providers’ obsession with control. But we should beware the haste with which some would sacrifice flexibility for control. If we can’t at least temper this taming of the chaotic PC, the victims will be competition, innovation and consumer freedom.¹²³

It is possible that security improvements will provide a useful pretext for software developers to impose restrictions on end-users motivated less by security than by the protection of copyrighted content from would-be copyright infringers¹²⁴ or the opportunity to impede competitors.¹²⁵

These are useful warnings about the possible outcome of demands for increased security in software design. Continued vigilance will be necessary to forestall, as far as is possible, this outcome. A preferable outcome of demands for improved security would be the adoption of better standards of software design.

★

4. LIABILITY IN NEGLIGENCE FOR UNREASONABLY INSECURE SOFTWARE

HAVING CONCLUDED THAT IT WOULD BE USEFUL to apply pressure to software vendors to improve the security of software, the question arises as to how this might be achieved. It would be possible for purchasers of products to sue software vendors for harm caused by their products. However, various impediments may prevent the purchaser from suing the software vendor. Licence terms disclaiming or limiting liability may discourage lawsuits. Furthermore, purchasers are open to counterclaims of contributory negligence if they have not been conscientious in maintaining the security of their own systems through patching and virus scanning. Finally, to the extent that software purchasers are harmed due to the insecurity of the software, the harm does not include the costs of DDOS attacks and so the incentives for purchasers to sue will be inadequate, at least at the margins.

121. Zittrain, *supra* note 59.

122. *Ibid.*

123. *Ibid.*

124. Richardson, *supra* note 22 at 14.

125. CCA, *supra* note 62 at 17–18.

The party who is best situated to sue a software vendor is the target of a DDOS attack. The target is likely to be reasonably large and to have suffered substantial losses that may make a tort lawsuit worthwhile. Two key problems confront such a suit. First, the victim of the DDOS attack (the “plaintiff”) must establish that the software vendor owed it a duty of care.¹²⁶ Second, the loss confronting the plaintiff in a DDOS attack is most likely to be “pure economic loss,” a category of loss for which courts have historically been reluctant to award damages. These two difficulties will be discussed briefly in turn.

4.1. Duty of Care

The Canadian test (often known as the *Anns* or the *Anns/Kamloops* test)¹²⁷ for the existence of a duty of care involves two stages. At the first stage, a court must determine whether there was a relationship of proximity between plaintiff and defendant and whether the harm to the plaintiff was a reasonably foreseeable consequence of the defendant’s act. If proximity and foreseeability are established, a *prima facie* duty of care is established and the court moves to the second stage of the *Anns* test. At the second stage, the court must determine whether there are residual policy considerations that suggest that a duty of care ought not to be recognized in the circumstances.

The Supreme Court of Canada has indicated that “proximity” is a concept used to identify those types of relationships in which a duty of care to guard against foreseeable harm may be imposed.¹²⁸ The Court has also noted that whether or not sufficient proximity exists in a novel situation is to be determined by reference to existing categories of relationships in which proximity has been found.¹²⁹ However, the list of such categories is not closed, and new categories may be recognized as analogous to existing categories. This permits the law of negligence to evolve to meet new circumstances. The Supreme Court’s discussion of proximity in *Cooper v. Hobart*¹³⁰ provides additional guidance on the characteristics that typify a relationship of proximity.

Defining the relationship may involve looking at expectations, representations, reliance, and the property or other interests involved. Essentially, these are factors that allow us to evaluate the closeness of the relationship between the plaintiff and the defendant and to determine whether it is just and fair having regard to that relationship to impose a duty of care in law upon the defendant.¹³¹

Having established both proximity and foreseeability of harm, a court moves to the second stage of the *Anns* test. At this second stage, the court con-

126. The basic elements of a negligence lawsuit in tort are (1) a duty of care owed by the defendant to the plaintiff, (2) a breach of that duty of care (i.e., failure to abide by a reasonable standard of care), and (3) the breach caused actual harm to the plaintiff.

127. See *Cooper v. Hobart*, 2001 SCC 79, < http://www.lexum.umontreal.ca/csc-scc/en/pub/2001/vol3/html/2001scr3_0537.html>, [2001] 3 S.C.R. 537.

128. *Ibid.* at 552.

129. *Ibid.* at 551.

130. *Ibid.*

131. *Ibid.* at 552.

siders whether residual policy concerns existing outside the relationship between the parties suggest that a duty of care should not be imposed. These concerns include the effect on other legal obligations, the legal system and society more generally of recognizing the duty of care.¹³² The existence of other remedies and the danger of “unlimited liability to an unlimited class” are relevant considerations at this stage.

In the DDOS context, then, it is necessary for the plaintiff to establish that there exists a relationship of proximity between itself as the target of a DDOS attack and the developer of insecure software that is exploited to launch the attack. The plaintiff must also establish the foreseeability of harm, and must show that there are no residual policy reasons not to recognize a duty of care.

As a preliminary matter, Canadian tort law already recognizes situations in which a defendant owes a duty of care to protect the plaintiff against the unlawful acts of a third party over whom the defendant has no direct control or authority.¹³³ For example, liability has been imposed in Canada on landlords whose inadequate security measures expose tenants to harm by third parties.¹³⁴ Liability is also imposed on schools, parents, physicians, youth and social workers for failing to protect children from physical or sexual abuse by third parties.¹³⁵ Liability has also been considered in situations where a defendant leaves the keys in a vehicle that is subsequently stolen by a third party who uses it to cause harm to the plaintiff’s person or property.¹³⁶ These cases stand for the proposition that a defendant may be liable in negligence where the defendant creates an unreasonable risk of damage by an intervening third party, which the defendant should have foreseen and guarded against.

In the DDOS context, the analogous argument would be that a duty of care to release reasonably secure software should be imposed on software developers, and that plaintiffs harmed when a third party takes advantage of unreasonably insecure software should be entitled to pursue the software developers in negligence for having created an unreasonable risk of harm. As noted above, the possibility of liability for creating an unreasonable risk of harm to the plaintiff at the hands of a third party is not a novel suggestion.

With respect to proximity, this paper suggests that where the vendor of a piece of software known to be a fundamental component of the internet (such as operating systems or browsers) has near complete market share, courts should find that there is a relationship of proximity between that vendor and all participants in the network. In a situation of near-monopoly, a vendor must know that its software in large measure creates the structure of cyberspace. Such a vendor must know that insecurities in its software expose all networked computers to attack by third parties. To borrow an analogy from the landlord liability cases, a

132. *Ibid.* at 554.

133. Robert M. Solomon, R.W. Kostal & Mitchell McInnes, *Cases and Materials on the Law of Torts*, 6th ed. (Scarborough: Thomson Carswell, 2003) at 288–290.

134. This duty is now usually grounded in occupier’s liability legislation. See e.g., *Allison v. Rank City Wall Can.* (1984), 29 C.C.L.T. 50 (Ont. H.C.J.); *Q. v. Minto Management* (1985), 49 O.R. (2d) 531 (H.C.J.), *aff’d* (1986), 57 O.R. (2d) 781 (C.A.).

135. *Supra* note 133 at 289 (footnote 9).

136. See e.g., *Spagnolo v. Margesson’s Sports* (1983), 41 O.R. (2d) 65 (C.A.); *Hewson v. Red Deer* (1976), 63 D.L.R. (3d) 168 (Alta. T.D.).

monopoly or near-monopoly software vendor has power over the characteristics of cyberspace in the same way as a landlord or occupier has power and authority over the characteristics of certain physical spaces. In other words, the law should recognize a form of “cyber-proximity”¹³⁷ in circumstances where a critical piece of software is so ubiquitous that it can be considered to determine the structure of cyberspace for everyone. Arguably, there is a relationship of proximity under these circumstances between the vendor of this software and all “inhabitants” of cyberspace.

The risk of harm due to insecure software is clearly foreseeable. Extensive experience with malicious code and cyber attacks that exploit software vulnerabilities has already accrued, making it inescapably foreseeable that software vulnerabilities produce the risk of substantial harm throughout the internet. In the case of a critical software program with near complete market share, the increased risks flowing from software homogeneity have already been noted. As a result, the particular ubiquity of the software not only increases risks and foreseeability of harm, but, as noted earlier, it also cements the relationship of proximity between the software vendor and the “inhabitants” of cyberspace.

The second stage of the *Anns* test, as outlined in *Cooper v. Hobart*, focuses on the existence of policy reasons not to find a duty of care. One possible policy concern is that the duty of care to all “inhabitants” of cyberspace exposes a software vendor to an indeterminate and huge number of possible plaintiffs, and unlimited potential damages. However, reasonable defences are available against many classes of plaintiff. Computer owners who fail to apply patches and to scan their systems for viruses may be held to have been contributorily negligent. Those who fail to shield critical functions from the internet environment, which is known to be insecure, may find it difficult to persuade a court that the software vendor should be responsible for enormous losses particularly compared to the relatively modest price of the software at issue. It is the victims of DDOS attacks who are using the internet for functions that can reasonably be entrusted to it (e.g. electronic commerce), and who have no reasonable means of self-protection, who are best-placed to sue software vendors in negligence.

There may in fact be strong policy reasons to find a duty of care in these circumstances. The internalization of the cost of DDOS attacks would likely induce greater attention to security during software development. To the extent that it is cheaper in the end to invest in security during the software development phase than to apply security measures once the software has been widely deployed, the imposition of liability on the software vendor would be an efficient outcome. Imposition of negligence liability would therefore not merely transfer wealth, but would produce gains in total social welfare. Furthermore, greater attention to security at the software development stage would have the positive side effect of reducing insecurity generally even though lawsuits are most likely to be successful only in the context of DDOS attacks. This is because the same types of vulnerabilities that lead to DDOS attacks also lead to other nuisances such as the circulation of malicious code in general.

137. The author is indebted to Dean Bruce Feldthusen of the University of Ottawa Faculty of Law for suggesting this term.

Furthermore, there is no other realistic remedy open to the targets of DDOS attacks. It is difficult to trace an attack to its source, meaning that lawsuits against ISPs, conscripted “zombie” attack computers and their owners, and against the criminal masterminds of the DDOS attacks are unlikely to succeed at present.

4.2. *The Harm of DDOS Attacks: Pure Economic Loss?*

The next difficulty facing a DDOS victim who wishes to sue a software vendor is that the losses seem largely to be pure economic losses, the recovery of which courts have traditionally been reluctant to permit.

Pure economic loss is economic loss unrelated to personal injury or property damage. Until fairly recently, courts have tended to refuse claims for the recovery of pure economic loss.¹³⁸ The reasons for judicial caution were listed by the Supreme Court in *Martel Building Ltd. v. Canada*:

To a large extent, this caution derives from the same policy rationale that supported the traditional approach not to recognize the claim at all. First, economic interests are viewed as less compelling of protection than bodily security or proprietary interests. Second, an unbridled recognition of economic loss raises the spectre of indeterminate liability. Third, economic losses often arise in a commercial context, where they are often an inherent business risk best guarded against by the party on whom they fall through such means as insurance. Finally, allowing the recovery of economic loss through tort has been seen to encourage a multiplicity of inappropriate lawsuits.¹³⁹

The Supreme Court of Canada has adopted a taxonomy developed by Professor Feldthusen,¹⁴⁰ which classifies recoverable pure economic loss into five categories.¹⁴¹ Nevertheless, the categories of recoverable pure economic loss are not closed.¹⁴² In *Martel Building Ltd.*, the Supreme Court indicated how courts should handle novel claims for the recovery of pure economic loss. In essence, Canadian courts are to apply the *Anns* test to determine whether or not to recognize a duty of care in the context of a novel claim for the recovery of pure economic loss.¹⁴³

The five recognized categories of recoverable economic loss¹⁴⁴ do not seem to be applicable in the context of a claim by a DDOS victim against the vendor of insecure software located on the “zombie” computers later used by a third party to launch the DDOS attack. As a result, it is necessary to consider the advisability of recognizing a new category of recoverable pure economic loss. The first stage of the *Anns* test was discussed above, and will not be repeated

138. *Martel Building v. Canada*, 2000 SCC 60 at para. 36, <http://www.lexum.umontreal.ca/csc-scc/en/pub/2000/vol2/html/2000scr2_0860.html>, [2000] 2 S.C.R. 860. “[T]he common law traditionally did not allow recovery of economic loss where a plaintiff had suffered neither physical harm nor property damage.”

139. *Ibid.* at para. 37.

140. Bruce Feldthusen, “Economic Loss in the Supreme Court of Canada: Yesterday and Tomorrow” (1991) 17:3 *Can. Bus. L.J.* 356.

141. *Martel Building Ltd.*, *supra* note 138 at para. 38.

142. *Ibid.* at para. 45.

143. *Ibid.* at paras. 46–47.

here. However, in *Martel Building Ltd.*, the Supreme Court gives additional guidance on the kinds of policy considerations relevant to the second stage of the *Anns* test when the recoverability of pure economic loss is at issue.¹⁴⁵ These policy considerations include (1) the danger of indeterminate liability (both in terms of the quantum of damages and the number of possible plaintiffs), (2) whether there is net social harm as a result of the activity, or simply the transfer of wealth between parties, (3) whether liability would deter socially and economically useful conduct or whether it would encourage economically efficient conduct, (4) whether liability would provide after-the-fact insurance to a plaintiff against a failure to act with adequate diligence to protect its own interests, (5) whether liability would require the courts to adopt a significant regulatory function, and (6) whether liability would encourage a multiplicity of lawsuits.

As noted earlier, indeterminacy may appear to be a problem with recognizing a duty of care in this situation. However, it is likely that only the targets of DDOS attacks would be successful (rather than the multitude of users of the insecure software), and while this is a theoretically large number, only those who suffer serious and sustained attacks causing major losses would be likely to sue. Furthermore, the quantum of damages can also be constrained to some extent where a target has foolishly entrusted mission-critical functions to internet-connected systems, which are known to give rise to certain risks.

With respect to the issue of whether the impugned activity generates a net social harm, a finding of liability in the DDOS context would likely improve the security of software, leading to reductions not only in DDOS attacks but also in losses due to the malicious code generally. As long as these savings exceed the costs of improved security, a net social gain will result. As noted earlier in the discussion, it is likely that this will be the case in the future if it is not already so.¹⁴⁶

Since the target of a DDOS attack can do little to avoid it, permitting recovery does not provide the plaintiff with protection against its own failure to undertake reasonable self-protection.

Although liability for unreasonable software insecurities would involve the courts in the technical area of software engineering standards, this is not more complex than the adjudicative tasks routinely undertaken by courts in other negligence cases involving sophisticated products.

Although a thorough review of the topic is beyond the scope of this paper, it would be interesting to explore the recent line of American cases dealing with trespass to chattels on-line.¹⁴⁷ These cases suggest that unauthorized electronic contact with computer systems may be actionable as a trespass to chattels, where the contact causes physical damage or impairs the functioning of the computer system. This line of cases may provide some means by which to argue that loss of use, without permanent physical damage, ought to be recov-

144. *Ibid.* at para. 38. These categories are the independent liability of statutory public authorities, negligent misrepresentation, negligent performance of a service, negligent supply of shoddy goods or structures and relational economic loss.

145. *Ibid.* at para. 61 ff.

146. See *supra* note 93 and accompanying text.

147. See e.g. *Intel Corporation v. Hamidi*, 30 Cal.4th 1342, <<http://www.law.com/regionals/ca/opinions/supremecourt/appeal01/c033076.shtml>>, 71 P.3d 296 (Supreme Court of California, 2003).

erable. As far as the author can determine, Canadian courts have not yet considered this line of American cases. Furthermore, it is not clear whether this line of cases is useful in the DDOS context. As Professor Radin points out, the trespass to chattels cases are not directly analogous to the DDOS situation.¹⁴⁸ In the trespass to chattels context, the plaintiff is suing the defendant trespasser, whereas in the DDOS context, the plaintiff would be suing someone else for negligently enabling the activities of the trespasser. However, it might be possible to craft an argument that a software vendor negligently created the risk that a third party would cause recoverable harm through trespass to chattels, a harm for which the vendor ought to be responsible, should it actually take place.

*

CONCLUSION

THE COSTS OF CYBER INSECURITY ARE HIGH, consisting not just of the actual losses but also of the lost opportunities of harnessing the efficiencies and benefits of the internet through increased use. Legislators, and to a lesser extent courts, have begun to experiment with different ways of encouraging improvements in various aspects of cyber security. This paper has focused on DDOS attacks, which are attacks that exploit insecurities in software in order to create an army of “zombie” computers with which to attack a target. Relying upon the assumption that it would be cheaper to improve software than to apply remedies and defences after software has been widely deployed to unsophisticated users, this paper discusses means by which software developers can be encouraged to adhere to secure software design standards. In particular, one promising means of encouraging better security is for the targets of such attacks to sue the vendors of insecure software for negligently creating an unreasonable risk of harm from third parties. Not only would such an approach help to prevent DDOS attacks, but it would also help to reduce the harms caused by malicious code more generally.

148. Margaret Jane Radin, “Distributed Denial of Service Attacks: Who Pays?” *Mazu Networks*, <http://www.mazunetworks.com/white_papers/radin.html>.