THE LAW AND ECONOMICS OF SOFTWARE SECURITY

ROBERT W. HAHN* AND ANNE LAYNE-FARRAR**

---

* Executive Director, AEI-Brookings Joint Center for Regulatory Studies.
** Director, LECG, LLC.

We would like to thank Brian Farrar, Keith Hylton, Larry Lessig, Douglas Lichtman, Ira Rubinstein, and Scott Wallsten for helpful suggestions, and Nadia Hussaini for research assistance. This research was supported by the AEI-Brookings Joint Center for Regulatory Studies and Microsoft. The views in this paper solely reflect those of the Authors and do not necessarily represent those of the institutions with which they are affiliated.

INTRODUCTION

Security in software networks relies on a complex mixture of technology, law, and economics. The considerable press surrounding security issues, the spread of worms and viruses on the internet, the possible link between identity theft and terrorism, and the penetration of online financial databases, attests to the subject's growing significance.

As the costs of software security breaches become more apparent, there has been a greater interest in developing and implementing solutions for different aspects of the problem. For example, the information technology community is prodigiously developing new fixes, ranging from gate-keeper protections to procedures for constructing more secure software. Increasingly, the federal government is paying more attention to this issue, particularly in the realm of online terrorism.[1] Additionally, there are numerous pending bills that would increase penalties for different kinds of cyber crime.[2]

Scholars address the software security problem from several different angles.[3] Most research in this area, however, focuses on discrete elements of the problem. Some scholars selectively focus on technical fixes that could help alleviate the problem,[4] whereas others examine the underlying institutions and incentives that shape consumer, business, and government responses. For example, Professor Randal Picker considers the

1. *See* Cyber Security Research and Development Act, Pub. L. No. 107-305, 116 Stat. 2367 (codified as amended at 15 U.S.C. §§ 7401–7411 (2006)) (discussed *infra* Part III.B).

2. *See infra* Part III.B.

3. For a review of the literature, see *infra* Part I.A. *See also* THE LAW AND ECONOMICS OF CYBERSECURITY (Mark F. Grady & Francesco Parisi eds., 2006) (serving as an overview of the various scholarly approaches).

4. *See, e.g.*, Neal Kumar Katyal, *Digital Architecture as Crime Control*, 112 YALE L.J. 2261, 2262 (2003).

issue from a structural point of view, asking whether a technological "monoculture" really weakens security.[5] He concludes that the security offered by having different technological platforms is not necessarily greater; indeed, sometimes the a diversity of platforms can create serious problems of its own.[6] In contrast, Douglas Barnes examines how policymakers could reduce the prevalence of viruses and worms by "deworming" the internet.[7] He suggests assigning some liability to both software developers and software users.[8] Finally, Kevin Pinkney analyzes how to overcome what he views as software developers' failure to provide secure code.[9] He too would assign some liability to developers but would allow ex post corrections to mitigate that liability.[10]

Although most research in this area is focused on discretely embedded elements, the security problems dealt with are not precisely defined, and researchers assume the problems are already well understood.[11] Similarly, many articles presume the particular issue they address is a serious problem in economic terms without specifically considering the total quantitative losses in more than a few incidents.

This Article seeks to address these gaps by presenting a comprehensive assessment of the software security issue using a law and economics framework. We begin by providing a

---

5. *See* Randal C. Picker, *Cyber Security: Of Heterogeneity and Autarky*, *in* THE LAW AND ECONOMICS OF CYBERSECURITY, *supra* note 3, at 115, 119 ("According to the [monoculture] claim, the Microsoft operating system monopoly creates a harmful monoculture—a common code base through which computer viruses spread easily putting the computing network at risk . . . . I believe that forced heterogeneity would be quite expensive and that we would be better off focusing on autarky, meaning here the conditions under which individual computers or internal systems within a firm should be isolated from the rest of the public network.").

6. For example, the monoculture idea "completely ignores consumer demand." *Id*. at 123. Moreover, "heterogeneity isn't equivalent to redundancy," meaning that when one system of a heterogeneous group fails, that system's users may still be without service because of the interrelated nature of the network. *Id*. at 125.

7. *See* Douglas A. Barnes, Note, *Deworming the Internet*, 83 TEX. L. REV. 279, 322–29 (2004).

8. *See id*.

9. Kevin R. Pinkney, *Putting Blame Where Blame is Due: Software Manufacturer and Customer Liability for Security-Related Software Failure*, 13 ALB. L.J. SCI. & TECH. 43, 62–82 (2002).

10. *See id*. at 69–82.

11. *See, e.g.,* Picker, *supra* note 5, at 116–18 (providing a useful categorization (cyber vandalism, cyber crime, and cyber terrorism), but not listing in any detail what kinds of attacks fall in each of the categories).

definition of software security that illustrates the complexity of the problem. We then review and critique the literature that assesses the costs of software security. Finally, we evaluate a number of possible approaches for addressing security problems using a law and economics framework.[12]

Our analysis leads to four key findings. First, software security problems come in many different shapes and sizes; therefore, the appropriate solutions will depend on the nature of the problem. Second, although attacks are becoming more common, the available data does not clearly establish that each aspect of software security poses a significant problem in terms of the damages inflicted by a breach. Some problems impose large costs on different groups, both in preventive and corrective costs. Other problems appear to function as more of a nuisance. Third, contrary to the prevailing view that market failures in the provision of software security are serious, some software users, particularly businesses, may face fairly strong incentives to take reasonable precautions. In response to this demand, several innovative market-based solutions have emerged to address a number of software security problems. Fourth, although some of the regulatory proposals for addressing security may be worth considering, most would require modification to ensure they do more good than harm. Moreover, broad interventionist proposals are difficult to justify given our findings about market-led responses. Instead, we conclude that the best role for the government would be to encourage the collection of more detailed data used to better inform policymakers on the need for specific actions. Furthermore, government agencies should seek to optimize their own security.

In Part II, we examine different aspects of software system security. In addition to defining software system security, we also consider the characteristics of different varieties of cyber criminals and their motivations for breaking into computer systems. Part III examines the economics of software system security. We provide an economic framework for evaluating software system security and assess the size of security related problems. We then review the underlying market failures that

12. *See, e.g.*, RICHARD POSNER, ECONOMIC ANALYSIS OF LAW (2003); John Prather Brown, *Toward an Economic Theory of Liability*, 2 J.L. STUD. 323 (1973). This Article also relies on cost-benefit analysis in assessing policy alternatives. *See, e.g.*, CASS SUNSTEIN, THE COST-BENEFIT STATE (2002).

contribute to software security problems. Part IV surveys the legal rules that apply to software. We also consider existing legislative efforts, and discuss whether those endeavors have been successful in addressing the known market failures. Finally, Part V analyzes policy proposals aimed at increasing software system security.

I. AN OVERVIEW OF SOFTWARE SECURITY

A. What is Software System Security?

Before assessing policies for addressing software security, it is important to have a clear understanding of what is meant by "software system security." The aim of software security is to reduce certain forms of damages.[13] Thus, one way of categorizing software security is through the types of damages caused by particular security breaches. Typically, there is an attacker, a method of attack, and the resulting damages. Table 1 provides an overview of different kinds of attacks and damages.[14]

Table 1: Categorizing Software System Security Problems

Key Types and Methods of Attack	**Types of Damage**
- Denial of Service	- Website Defacement and Outages
- Viruses, Worms, and Trojan Horses	- Data and Software Corruption
- Exploitation of Trust through Social Engineering (e.g., Phishing)	- Privacy Breaches of Databases and Personal Records
- Computer Spying (e.g., Spyware)	- Identity Theft
- Exploitation of Program Vulnerabilities (e.g., Backdoors)	- Financial Fraud
	- Misuse of Trade Secrets or State Secrets

13. We will examine the relationship between attacks and damages more carefully later when we explore the quantitative data on damages. *See infra* Part III.B. We also consider the relative merits of different solutions to software security problems by weighing whether proposed solutions reduce the damages caused by an attack or lower the cost of preventing such attacks. *See infra* Part V.

14. The table is by no means exhaustive, as types and methods of attack are continually evolving. Moreover, as further discussion clarifies, the types of attack are not mutually exclusive. Some attacks blend a number of approaches.

1. Types and Methods of Attack

All of the items listed under the first column, Key Types and Methods of Attack, involve approaches intended to breach a computer network's defenses. The table reveals that there are many different routes for attacking computers or networks, many of which may be combined in a single attack.

In a denial of service attack, a network is inundated with worthless traffic, overwhelming the network and shutting down access.[15] Although the traffic itself may not appear unauthorized, "the volume and frequency of the traffic will increase to unmanageable levels."[16] If, for example, an internet access provider like AOL were hit with a successful denial of service attack, all AOL subscribers would be unable to sign into their accounts, read their emails, or gain access to the internet. These attacks were particularly popular during the early days of the internet when a single large email attachment could bring down a network.[17]

More recently, worms and viruses[18] have received considerable press, particularly the notorious Love Bug and Blaster worms.[19] These self-replicating programs are typically sent through email, often corrupting data files and programs on a

---

15. This category includes distributed denial of service attacks, in which multiple computers are deployed in an attack. *See* Webopedia, Denial of Service (DoS) Attacks, http://www.webopedia.com/DidYouKnow/internet/2005/DoS_attack.asp (last visited Sept. 30, 2006) (explaining denial of service attacks).

16. *Id.*

17. *See generally* Kevin J. Houle & George M. Weaver, *Trends in Denial of Service Attack Technology* (CERT Coordination Ctr. Paper, 2001), *available at* http://www.cert.org/archive/pdf/DoS_trends.pdf.

18. A worm is "[a] program or algorithm that replicates itself over a computer network and usually performs malicious actions, such as using up the computer's resources and possibly shutting the system down." Webopedia, Worm, http://www.webopedia.com/TERM/w/worm.html (last visited Sept. 30, 2006). The terms "viruses" and "worms" are sometimes used interchangeably, although some people distinguish between the two: "A worm is a special type of virus that can replicate itself and use memory, but cannot attach itself to other programs," whereas a virus is "[a] program or piece of code that is loaded onto [a] computer without [the user's] knowledge and runs against [his] wishes" with the ability of self-replication. Webopedia, Virus, http://www.webopedia.com/TERM/v/virus.html (last visited Sept. 30, 2006).

19. *See* Will Sturgeon, *Love Bug Legacy Shows Users Still Fall for False Promises*, SILICON.COM, May 4, 2005, http://software.silicon.com/security/0,39024655, 39130122,00.htm; Sheryl Silverman, *Famous Worms and Viruses*, ONLINE NEWSHOUR, Dec. 1, 2003, http://www.pbs.org/newshour/science/computer_worms/famous.html.

recipient's personal computer.[20] The Love Bug worm appeared in Asia in May 2000 and quickly spread to the United States through email attachments, affecting government computers at Congress, the White House, and the Pentagon.[21] By some estimates, this worm caused $10 billion in economic damages by overwriting files and corrupting data.[22] Three years later, in August 2003, the Blaster worm appeared and exploited vulnerabilities in operating systems.[23] Although its effects were far reaching, they appear to be less costly than those of the Love Bug.[24] The Blaster worm slowed down personal computer response times, with some machines requiring a reboot to restore operations.[25] According to a survey by the International Data Corporation (IDC), a leading technology research firm, viruses and worms are the "most serious threat facing corporations today . . . ."[26] Because so many people are affected by this type of attack, worms and viruses are especially visible to the public, the press, and policymakers.[27]

Trojan horses are a type of malicious software (malware) related to worms and viruses.[28] Like their namesake from Greek mythology, these programs seem benign, but actually contain malicious code.[29] Although some Trojan horses merely change simple desktop settings, others can cause serious damage by deleting files and destroying information.[30] The MyDoom Trojan horse of 2004 opened a backdoor that enabled its author to download personal data from infected computers and caused an estimated $4.8 billion in damages.[31] Trojan horses generally

---

20. *See generally* Silverman, *supra* note 19.

21. *See id.*

22. Thomas McCann, *The Virus War: Plugging Holes*, CHI. TRIB., Aug. 21, 2000, § 4, at 1.

23. *See* Silverman, *supra* note 19.

24. *See id*.

25. *See id*.

26. BRIAN E. BURKE ET AL., IDC, WORLDWIDE IT SECURITY SOFTWARE, HARDWARE, AND SERVICES 2005-2009 FORECAST: THE BIG PICTURE 2 (2005).

27. *See Fighting the Worms of Mass Destruction*, ECONOMIST, Nov. 29, 2003, at 65.

28. *See* Webopedia, Trojan Horse, http://www.webopedia.com/TERM/T/Trojan_horse.html (last visited Sept. 30, 2006).

29. *See id*.

30. *See* Webopedia, Destructive Trojan, http://www.webopedia.com/TERM/D/Destructive_Trojan.html (last visited Nov. 7, 2006).

31. Brian Grow, *Hacker Hunters*, BUS. WK., May 30, 2005, at 74, 78.

cannot replicate themselves,[32] and must therefore rely on malicious or unsuspecting end users to spread through a network.

Trojan horses are successful because of people's general inclination to trust, a trait also exploited by con artists using "social engineering."[33] The con artist in this case will trick someone into revealing otherwise secure information and thus enable network or database access that appears authorized but is not.[34] Social engineering is not a new problem,[35] but the emergence of the internet has only made it more prevalent. One growing form of online social engineering is fraudulent email, known as "phishing."[36] Unlike a worm or Trojan horse, phishing does not do any damage on its own, but instead helps to feed confidential data to criminals who then use it for theft and fraud.[37] With phishing, the malicious hacker poses as a legitimate business, like Amazon.com or Chase Manhattan Bank, by using an email that looks authentic.[38] The email might link to the business's real website, but hackers may attach a password skimmer, which will allow the hacker to later return to collect the user's personal information. Alternatively, the email may direct a victim to a false website that is carefully crafted to resemble the real company's site.[39] Once there, the victim is instructed to enter personal and financial data, such as passwords and social security numbers, to "update" their records.[40] According to one research firm, phishing scams cost banks and credit-card issuers more than $1.2 billion in 2004.[41]

Other forms of social engineering take advantage of well-known human behaviors. If individuals, either as employees or as consumers, are careless with security measures, malicious hackers can access software files or networks by posing as le-

---

32. Webopedia, Trojan Horse, *supra* note 28.

33. *See* Webopedia, Social Engineering, http://www.webopedia.com/TERM/s/social_engineering.html (last visited Oct. 2, 2006).

34. *See id.*

35. The popular movie CATCH ME IF YOU CAN (Dreamworks 2002), starring Leonardo DiCaprio and Tom Hanks, portrayed the true story of social engineering perpetrated by Frank Abagnale, Jr. in the 1960s.

36. *See* Webopedia, Social Engineering, *supra* note 33.

37. *See* Webopedia, Phishing, http://www.webopedia.com/TERM/p/phishing.html (last visited Oct. 2, 2006).

38. *See id*.

39. *See id*.

40. *See id*.

41. Mara Der Hovanesian, *Hackers and Phishers and Frauds, Oh My!*, BUS. WK., May 30, 2005, at 81.

gitimate users. For example, a user might select an easy-to-remember word as a password rather than a combination of letters and numbers. Knowing this, a hacker could use a simple program to generate a list of possible passwords until he gains access to the system.[42] Moreover, even if the user followed recommended protocols for selecting passwords, he might keep a list of passwords in a location easily accessible to malicious hackers.[43] In some instances, end users may simply share passwords without considering or understanding the consequences. A British survey revealed that ninety percent of respondents exchanged their office computer password for an inexpensive pen.[44] Malicious hackers exploit these tendencies, posing as trusted parties in order to gain enough information to break into an otherwise secure network.[45]

Spyware represents another fraudulent route to collecting personal data. These programs are installed on users' computers without their knowledge, often because the spyware is hidden in shareware or peer-to-peer files.[46] Once installed, the programs surreptitiously collect information, including internet sites visited, financial data, and passwords.[47] IDC estimates that as many as three-quarters of all corporate machines are infected with some form of spyware.[48]

Another route for attack is through software vulnerabilities. For example, an intentional vulnerability occurs when a software programmer places a "backdoor" in a program, which allows the programmer or his confederates to access the program without going through formal password protections and other security measures established at a later time.[49] Backdoors

---

42. *See, e.g.*, Stephen T. Irwin, *What Corporate Users Should Know About Data Network Security*, TELECOMM., May 1991, at 49.

43. *Id.*

44. *See* Peter J. Toren, *Limiting Computer Crime Losses with Cyberinsurance*, THE INTERNET NEWSLETTER, July 6, 2004, at 1 (on file with HARV. J. L. & PUB. POL'Y).

45. Kevin Mitnick, one of the nation's most infamous hackers, exploited social engineering tactics in order to bypass organizational security. *See generally* KEVIN D. MITNICK & WILLIAM L. SIMON, THE ART OF DECEPTION: CONTROLLING THE HUMAN ELEMENT OF SECURITY (2002).

46. *See* Webopedia, Spyware, http://www.webopedia.com/TERM/s/spyware.html (last visited Oct. 2, 2006).

47. *Id.*

48. BURKE ET AL., *supra* note 26, at 2.

49. A backdoor is "[a]n undocumented way of gaining access to a program, online service or an entire computer system. The backdoor is written by the programmer who creates the code for the program. It is often only known by the

can allow individuals to obtain unauthorized access, even after a program is licensed or sold to a third party. The classic example among IT professionals is a clever use of code that creates a backdoor and then hides any signs of its existence.[50] This form of security breach raises concern in the context of global software sales, where governments speculate whether software developed in foreign nations masks spying tools.[51]

The exploitation of software vulnerabilities does not, however, require the intentional inclusion of a backdoor. Unintentionally poor software design can also make it easier for outside parties to gain unauthorized access to a network or data files.[52] For instance, programs containing temporary storage areas called buffers can experience "buffer overflow" problems.[53] Sophisticated hackers are able to determine the length required to cause an overflow in a program.[54] They then write code to purposely exceed the buffer length, causing the submitted data to overflow into an adjacent buffer, where the extra data overwrites and thus corrupts a file, or where malicious instructions can be released and executed.[55] Even though programmers are now aware of the problem, problems of this sort can persist since software is typically written cumulatively, by combining new and old code.[56]

---

programmer." Webopedia, Backdoor, http://www.webopedia.com/TERM/b/backdoor.html (last visited Oct. 2, 2006).

50. *See* Ken Thompson, *Reflections on Trusting Trust*, *in* COMPUTERS UNDER ATTACK: INTRUDERS, WORMS, AND VIRUSES 97, 101–02 (Peter J. Denning ed., 1990) (reprint of a 1983 Assoc. for Computing Mach. (ACM) Turing Award lecture).

51. *See* Tim Lee, *Linux in the Red*, LINUX MAG., Feb. 11, 2005, *available at* http://www.linux-mag.com/content/view/41/112.

52. *See* Picker, *supra* note 5, at 137.

53. *See* Sandeep Grover, *Buffer Overflow Attacks and Their Countermeasures*, LINUX J., Mar. 10, 2003, *available at* http://www.linuxjournal.com/article/6701.

54. *See id*.

55. *See id*.

56. The SANS Institute, in cooperation with the National Infrastructure Protection Center, annually publishes a list of the 20 most commonly exploited vulnerabilities for Linux, Unix, and Windows systems. Software developers can use these lists to improve the security of their programs. *See* SANS Institute, Top 20 Internet Security Attack Targets (2006 Annual Update), http://www.sans.org/top20 (last visited Nov. 12, 2006). It is important to emphasize that both proprietary and open-source software are on this list and others like it. Attacks on less prevalent non-Windows platforms, such as Linux, Apple, and Unix, appear to be on the rise. *See* Jon Olstik, *Good Security News to be in Short Supply in 2006*, CNET NEWS.COM, http://news.com.com/2102-1071_3-6028980.html.

2. Types of Damage

The second column in Table 1 covers the potential harm to websites, networks, files, and data that can result from attacks. The first item listed, website defacements, is the online version of graffiti. Hackers may act as social activists, or "hacktivists,"[57] and change a website's appearance, add political messages to the site, or divert visitors to another website. Another form of website damage is an outage. This is typically the result of a successful denial of service attack and can be costly, especially for e-commerce sites.

Data corruption and unauthorized database access are particularly serious concerns. For instance, malware that overwrites files or erases databases can result in significant losses for businesses. This category also includes privacy breaches, where sensitive personal data, such as social security numbers or medical records, may be accessed inappropriately. For example, in 2005 alone, at least 17 U.S. universities suffered breaches to private servers that contained students' social security numbers.[58]

Data theft is one of the costliest types of damage resulting from a security breach.[59] When an unauthorized person accesses a database containing credit card numbers and financial account data, the breach can lead to credit card fraud and identity theft. The recent instance of data theft at ChoicePoint, a consumer data collection agency, illustrates the dangers inherent in this type of security lapse.[60] In that case, a Nigerian national living in California posed as the head of a business and duped ChoicePoint into granting him access to its data files. The perpetrator, in conjunction with other individuals, then used the data gleaned from the ChoicePoint files to establish

---

57. "[H]acktivism is the act of hacking into a Web site or computer system in order to communicate a politically or socially motivated message." Webopedia, Hacktivism, http://www.webopedia.com/TERM/h/hacktivism.html (last visited Oct. 5, 2006).

58. For a list of privacy breaches in 2005, see Identity Theft Resource Center, 2006 Disclosures of U.S. Data Incidents, http://www.idtheftcenter.org/breaches.pdf (last visited Nov. 17, 2006).

59. In 2003, the FBI reported that intellectual property theft amounted to a loss of $70.1 million. Duane Hopkins, Securing the Company's Intellectual Property, Oct. 5, 2005, http://www.intellectualsecurity.com/2005/10/securing_the_companys_intellec.html.

60. *See* Reuters, Man Faces New Charges in ChoicePoint ID Theft (Aug. 31, 2005), *available at* http://news.zdnet.com/2100-1009_22-5844937.html.

fraudulent credit card accounts. As a result, card issuing banks lost approximately $2 million and ChoicePoint incurred $2 million in costs associated with notifying customers.

Consumer financial information is not the only target of computer-savvy criminals. Indeed, corporations and governments are also vulnerable to the same threats of computer theft in the form of trade or state secrets. Corporate espionage frequently involves unauthorized access to computer systems, often by company insiders emailing files to outside parties.[61] Although the theft of trade and state secrets is not specific to software systems,[62] use of the internet greatly increases the ease with which the illegally-obtained information can be quickly and broadly disseminated, thus exponentially increasing the extent of the damage. For example, several recent trade secret cases have involved hackers posting sensitive materials on the internet, giving anyone with an internet connection the opportunity to access the information and use it to the detriment of others.[63] In late 2003, a malicious hacker broke into T-Mobile USA's computer systems and stole the names and social security numbers of about 400 customers.[64] After obtaining the information, the hacker attempted to sell it on an underground website, resulting in at least $5,000 worth of damages.[65]

B. Identifying Cyber-Criminals and Their Motivations

An analysis of the people behind cyber attacks, as well as their respective motivations, is useful in determining the appropriate

---

61. *See, e.g.*, Birgitta Forsberg, *The Spies in the Next Cube: Silicon Valley a Magnet for Trade Secret Theft—and It's Often an Inside Job*, S.F. CHRON., Apr. 25, 2005, at E1.

62. "Check and payroll fraud, stolen credit card numbers and PINs, e-mail 'phishing' scams, spyware on home computers, scanning computer monitors (with infrared readers or precision binoculars), and old fashioned dumpster diving" are all means of acquiring data for identity theft or trade secret theft. William A. McComas, *Weak Links: To Prevent Identity Theft, Congress Should Focus on Banking and Credit Card Industries*, LEGAL TIMES, Oct. 17, 2005, at 84. Although some of these methods may require the *indirect* use of computers, none of them involve directly hacking into an online database. *See id.*

63. *See, e.g.*, Religious Tech. Ctr. v. Lerma, 908 F. Supp. 1362 (E. D. Va. 1995); Religious Tech. Ctr. v. Netcom On-Line Commc'n Servs., 923 F. Supp. 1231 (N.D. Cal. 1995).

64. *See T-Mobile System Hacked in 2003*, SEATTLE TIMES, Jan. 13, 2005, at E1, *available at* http://archives.seattletimes.nwsource.com/cgi-bin/texis.cgi/web/vortex/display?slug=tmobile13&date=20050113.

65. *Id*.

policy responses. In this section, we discuss the people behind software system attacks and their apparent objectives.

The first perpetrator we examine is the cyber-terrorist. Cyber-terrorism has been defined as "a computer-based attack or threat of attack intended to intimidate or coerce governments or societies in pursuit of goals that are political, religious, or ideological."[66] Most computer security experts acknowledge that the threat of cyber-terrorism is quite real,[67] although it is difficult to differentiate these incidents from other kinds of software system attacks. The specific tools of attack employed by cyber-terrorists can be largely identical to those used by traditional hackers, including denial of service attacks, worms and viruses, and compromised insiders. Distinguishing between cyber-terrorism and run-of-the-mill hacking must therefore be done through an examination not of the methods utilized by the perpetrators, but rather of their targets and apparent intentions. Just as acts of physical terrorism are often aimed at structures that are critical to the infrastructure of society, most experts believe that cyber-terrorism would likely involve targets that are crucial to the 'electronic infrastructure' of society, such as emergency response services or electrical power grids.[68]

Although differences in targets are easy to predict, intent is notoriously difficult to determine. For example, in 1998, a round of serious cyber attacks aimed at the Pentagon were initially attributed to foreign terrorists, but investigations later determined that the attacks instead originated from three teenagers, two American and one Israeli.[69] Moreover, it took some time for investigators to make this determination and the attacks themselves lasted for nearly a month. The practical implication of this delay in identification of acts of cyber-terrorists is that malicious hacking will generally not be truly distinguish-

---

66. Dorothy E. Denning, Is Cyber Terror Next?, Nov. 1, 2001, http://www.ssrc.org/sept11/essays/denning.htm.

67. For a summary of the issues involved in assessing the threat of cyberterrorism, see GABRIEL WEIMANN, CYBERTERRORISM: HOW REAL IS THE THREAT? 1 (U.S. Inst. of Peace, Spec. Rep. 119, Dec. 2004), *available at* http://www.usip.org/pubs/specialreports/sr119.pdf. Weimann states that although "no single instance of real cyberterrorism has been recorded," the potential is considered very real. *Id.*

68. *See generally* Dorothy E. Denning, *Cyberterrorism: The Logic Bomb Versus the Truck Bomb*, GLOBAL DIALOGUE (Autumn 2000).

69. *See* Thomas Bello, *Cyber Security: Battleground of the Future?*, 14 SECURITY TECH. & DESIGN, Apr. 2004, at 42, *available at* http://www.securityinfowatch.com/article/article.jsp?id=906&siteSection=439 (discussing the "Solar Sunrise").

able from cyber-terrorism until well after the fact. The policy implication is that laws targeting cyber-terrorism are not likely to be as effective as laws that focus on the types of attacks and the damage they cause.

Malicious hackers can be sorted according to ability. Many people conjure up images of alienated teenage boys hacking for the thrill of it from the bedrooms of their suburban homes. There may be some truth to that image, but not much. The term "script kiddies" refers to relatively unskilled young hackers who deploy malicious hacking tools ("scripts") developed by others.[70] These individuals can play an important role in undermining security despite their lack of technical prowess, since they are the vehicles that skilled hackers rely upon for a broad deployment of their own malware. A report issued by IBM Global Security Analysis Lab in 2002 estimated that 90% of hackers are "amateurs with limited technical proficiency."[71] The 90% figure is accordingly dependent on a much smaller group of truly skilled programmers, estimated to be comprised of only 9% of the hacker community, to supply the tools the community uses to wreak havoc. The same study estimated that 9% of hackers "are more skilled at gaining unauthorized access, but do not damage the files they read . . . ."[72] Only 1% of the hackers are both highly skilled and possess malicious intent.[73]

Though small in number, this 1% has considerable reach in today's computing environment. Many software programs, particularly operating systems, are long, intricate, interrelated segments of code. For example, some estimates put Microsoft's Windows 2000 operating system at over 29 million lines of code.[74] Given the complexity of programs such as these, most experts believe that bugs and other vulnerabilities are inevita-

---

70. A "script kiddie" is "[a] person, normally someone who is not technologically sophisticated, who randomly seeks out a specific weakness over the internet in order to gain root access to a system without really understanding what it is s/he is exploiting because the weakness was discovered by someone else." Webopedia, Script Kiddie, http://www.webopedia.com/TERM/S/script_kiddie.html (last visited Oct. 31, 2006).

71. WEIMANN, *supra* note 67, at 9.

72. *Id.*

73. *Id.*

74. *See* Microsoft Windows 2000 Help, http://www.computerhope.com/win2000.htm (last visited Nov. 17, 2006).

ble.[75] Just as a skilled house burglar may be deterred by alarms and intrusion detection systems but with enough motivation can almost always find a way into the home, a skilled hacker who is fueled by sufficient determination to do harm will virtually always be able to penetrate a given system. Unfortunately for the victims of cyber break-ins, however, the similarity to real-world burglary ends there. Although the damages sustained by a break-in victim are usually limited to whatever the original burglar steals, victims of a hacking will often sustain damages far beyond those inflicted by the original hacker, due to the rapidity and ease with which news of breach can spread across the world wide web to others who can take advantage of it before it is discovered by the victim.[76]

Automated tools "are becoming increasingly powerful and easy-to-use, with graphical user interfaces that require little skill on the part of the user."[77] Cyber security expert Dorothy Denning has observed that "[f]or a few dollars, anyone can buy a disk with thousands of [computer viruses]. Alternatively, interested persons can download them and other types of cyberweapons from the internet. Typing 'hacking tools' into one internet search engine yielded 42,012 hits in March 2000 and 64,669 in July."[78] The availability of such tools continues to expand.[79]

Nor are easy-to-use hacking tools the only security threat that is proliferating. The number of highly skilled malicious hackers will likely grow over time as well. IDC observes that "[t]he evolution from mischievous hobby to moneymaking

---

75. *See, e.g.*, Reid Skibell, *The Phenomenon of Insecure Software in a Security-Focused World*, 8 J. TECH. L. & POL'Y 107, 110 (2003). Along these same lines, IDC observes that "as IT infrastructures become more complex, it is not possible to protect everything. Instead managers need to prioritize their security to protect the most critical assets." BURKE ET AL., *supra* note 26, at 4.

76. Some automated malware does not even require unskilled labor for propagation. With "zombies," attacks can take place without a computer's user/owner being aware his or her computer is launching attacks. A zombie is "[a] computer that has been implanted with daemon [a program that runs in the background] that puts it under the control of a malicious hacker without the knowledge of the computer owner. Zombies are used by malicious hackers to launch [denial of service] attacks." Webopedia, Zombie, http://www.webopedia.com/TERM/z/zombie.html (last visited Oct. 2, 2006).

77. Dorothy E. Denning, *Reflections on Cyberweapons Controls*, 16 COMPUTER SECURITY J. 43, 44 (2000).

78. *Id.* at 43.

79. Typing the same word string ("hacking tools") into the Google search engine in September 2005 retrieved a list over 11 million hits long.

criminal venture has attracted a new breed of sophisticated hackers and organized crime . . . his profit-driven motivation will cause the number of attacks to increase in sophistication, frequency, and severity."[80]

In addition to the highly skilled malicious hackers who create malware tools, disgruntled insiders can also pose a serious threat. Insider-assisted attacks are particularly problematic in larger organizations.[81] Reasons for purposeful inside attacks can range from profit to revenge.[82] Counting both purposeful and accidental attacks, one estimate reckons that "approximately 80 percent of reported network compromises are insider cases, in which there was some involvement on the part of an employee, former employee, contractor, vendor, or some other person who either holds or previously held some degree of trusted access status."[83] A survey on security issues among businesses and government agencies puts the proportion of insider-assisted breaches closer to fifty percent.[84] Since most security breaches are not reported to any authority, however, the percentage of attacks involving inside assistance is difficult to determine.

Our review of software system security and cyber criminals reveals that computer system security is a complex issue. Attacks can be accomplished by exploiting either a weakness in the software itself, a mistake made by its user, or a vulnerability on the part of an internet access provider. The people behind these attacks can be anonymous criminals, insiders intent on revenge, or amateur hackers using automated malware that they could not create on their own.

II. THE ECONOMICS OF SOFTWARE SYSTEM SECURITY

In this Part, we provide an economic framework for evaluating software security solutions. After defining an objective based

---

80. BURKE ET AL., *supra* note 26, at 3.

81. Roughly 45% of all security threats in large organizations are internal, compared to about 25% in small organizations. *See* IDC, WORLDWIDE OUTBOUND CONTENT COMPLIANCE 2005-2009 FORECAST AND ANALYSIS: IT SECURITY TURNS INSIDE OUT 9 (2005).

82. *See generally* STEVEN BRANIGAN, HIGH-TECH CRIMES REVEALED: CYBERWAR STORIES FROM THE DIGITAL FRONT (2004).

83. Bello, *supra* note 69; *see also* WEIMANN, *supra* note 67, at 9 ("For now, insiders intent or individual hackers are responsible for most attacks and intrusions.").

84. LAWRENCE A. GORDON ET AL., 2005 CSI/FBI COMPUTER CRIME AND SECURITY SURVEY 14 (2005) [hereinafter CSI/FBI 2005 SURVEY].

on minimizing prevention costs and damages, we assess the available data on the costs imposed by software system breaches. We then examine the underlying market failures that complicate attempts to address existing security problems.

A. A Framework for Evaluating Software System Security

From an economist's perspective, before the government decides to intervene to improve software security, it must be reasonably certain that private parties are unable to do so on their own. In other words, it must be clear that the market has failed in some way. Otherwise, interventions run the risk of interfering with properly functioning markets and, therefore, of introducing inefficiencies where none existed before—what could be termed a "government failure" as opposed to a market failure.

Once a market failure related to the provision of software security is identified, however, a feasible solution still must be developed. Spam provides an example of the difficulty in creating effective solutions. Congress has debated a number of proposals to address the ever-expanding issue of junk email.[85] In some instances, spam is merely a nuisance—unwanted email to delete from the inbox. In other instances, it has interfered with internet service providers' networks, slowing down or interrupting service altogether.[86] Some kinds of spam, like messages for pornography sites, can be highly offensive to many recipients. Despite these legitimate reasons for regulating unsolicited email, the practical reality is that today's technology provides little remedial relief. Legitimate marketers may abide by the rules to attach truthful subject lines and remove recipients upon request,[87] but the real offenders can easily evade any U.S. legislation by constantly moving their operations or by locating offshore in coun-

---

85. *See, e.g.*, Criminal Spam Act of 2003, S. 1293, 108th Cong. (2003); CAN-SPAM Act of 2003, S. 877, 108th Cong. (2003) (enacted); Anti-Spam Act of 2003, H.R. 2515, 108th Cong. (2003); Reduction in Distribution of Spam Act of 2003, H.R. 2214, 108th Cong. (2003); REDUCE Spam Act of 2003, H.R. 1933, 108th Cong. (2003); Netizens Protection Act of 2001, H.R. 3146, 107th Cong. (2001); Anti-Spamming Act of 2001, H.R. 1017, 107th Cong. (2001); Anti-Spamming Act of 2001, H.R. 718, 107th Cong. (2001); Wireless Telephone Spam Protection Act, H.R. 113, 107th Cong. (2001).

86. *See generally* BRIAN E. BURKE & ROSE RYAN, IDC, WORLDWIDE SECURE CONTENT MANAGEMENT 2005–2009 FORECAST UPDATE AND 2004 VENDOR SHARES: SPYWARE, SPAM, AND MALICIOUS CODE CONTINUE TO WREAK HAVOC (2005).

87. As mandated by the CAN-SPAM Act of 2003 § 5, 15 U.S.C. § 7704 (2003).

tries with little or no regulation.[88] An increasingly popular tactic among the more disreputable spammers is to take over thousands of individuals' personal computers, most often without the knowledge of the owner, and use them to send spam.[89] The spam then looks as if it were coming from numerous legitimate sources, making it harder for spam filters to identify. Thus, a viable solution to the market failure must be plausible before outside intervention is warranted. Without such practical consideration, legislation is likely to hamper legitimate players unnecessarily without solving the problem at hand.

After a solution is identified, the hardest part of policy analysis begins—assessing whether the benefits are likely to outweigh the costs. If a proposed solution is likely to cause more harm than good, then even in the face of a market failure, the option of doing nothing could still be the best course of action. Cost-benefit analysis requires an examination of a proposal's impact on incentives, along with its potential for unintended consequences.

One reasonable cost-benefit approach for software security would be based on an analogous issue in the physical world. The law and economics literature frequently frames physical accident regulatory issues as cost minimization problems.[90] Accidents result in damages, both physical and economic, such as the hospital bills a patient must pay for injuries sustained in a car wreck. Preventing accidents also involves costs, such as expenditures on safer car brakes or the discomfort involved in wearing a seat belt. There are also administrative costs associated with court proceedings and enforcing a particular regulatory regime, such as speed limits. Accordingly, the problem to be solved is one of minimizing the sum of accident costs, prevention costs, and administrative costs. For software security the problem is similar. Regulation should seek to minimize the sum of damage costs from security breaches, the costs of preventing security breaches, and administrative costs.

---

88. *See* Eugene Altovsky, Coordinator, AntiSpam Project, Russian Federation: No Regulation, Legislation Currently in Process, http://www.itu.int/osg/spu/spam/contributions/UNESCO-IFAP.pdf.

89. *See* BURKE & RYAN, *supra* note 86. A computer that has been "taken over" surreptitiously by a malicious hacker is called a "zombie." A network of zombie computers is referred to as a botnet, short for robot network.

90. *See generally* GUIDO CALABRESI, THE COST OF ACCIDENTS: A LEGAL AND ECONOMIC ANALYSIS (1970); Richard A. Posner, *Guido Calabresi's* The Costs of Accidents*: A Reassessment*, 64 MD. L. REV. 12 (2005).

One would like to be able to compare the costs and benefits of various systems to the extent the data permit. This Article introduces a stylized model that illustrates how this might be done. Suppose a business is deciding whether to invest in security. If it does not invest in security, we assume it incurs a loss L with some probability p and incurs an administrative cost C with the same probability. These administrative costs could arise from dealing with lawsuits, processing complaints, or negotiating settlements. The business's expected losses in this case are $pL + pC$.

Now, suppose the business can invest in security at a cost X, giving rise to some lower probability of loss q. For simplicity, we assume that the loss, L, and administrative cost, C, remain the same, although the investment might lower them as well. The firm's costs, if it chooses to make this investment, are $X + qL + qC$. It will make the investment in security (i.e., take care) if:

(1) $pL + pC > X + qL + qC$

We can think of this as the case of pure private enforcement. We have some preliminary data on qL and X that we present below, which allows us to illustrate how one might determine a lower bound for the left hand side of (1).

This model can be extended to consider the costs and benefits of government intervention. Suppose the government were considering introducing some kind of deterrence mechanism. This mechanism could take the form of introducing fines, jail terms, or information-sharing that lowered the cost of detection. Say the new mechanism results in administrative costs equal to G (for each business) and they lower the probability of loss from q to r ($p > q > r$). For simplicity, assume that (1) is satisfied, so the firm decides to invest in security. The government should then invest an additional amount G per business if it further lowers the expected cost. That is, if

(2) $X + qL + qC > X + G + rL + rC$

One can think of (2) as a mix of private and public enforcement.[91] In this case, government enforcement may be useful be-

91. *Cf.* Steven Shavell, *The Social Versus the Private Incentive to Bring Suit in a Costly Legal System*, 11 J. LEGAL STUD. 333 (1982) (explaining a related economic model).

cause private individuals may not supply the optimal level of enforcement.[92] One option for the government in this case would be to lower the penalty in order to give victims a greater incentive to invest in security.[93] The desirability of government intervention in (2) depends on whether an increase in government expenditures lowers the probability of loss sufficiently to make the investment worthwhile. More generally, government will want to invest if its investments yield changes in q, L, or X that make the government investment worthwhile.[94]

The foregoing economic model suggests that there are at least two approaches for dealing with software security. One is ex ante investment in security for software developers, internet access providers, consumers, and businesses. The other is ex post punishment through liability for any of the above parties not taking "adequate" precautions, or through fines or prison sentences for hackers.[95]

B. The Economic Costs and Damages Involved

Software and network security issues receive substantial press,[96] but it is difficult to determine the extent of economic harm resulting from these issues simply by reading news reports. In this section, we analyze the available data for measuring the economic harm generated by software system security breaches. We find that the level of genuine harm in the economy varies dramatically by the type of security breach, but the available data leave much to be desired.

1. Measuring the Loss

Damages fall into one of several overlapping categories. First, an attack can harm a computer system, such as taking a network offline or erasing data. Restoring a network or recreating the lost data often imposes heavy quantifiable costs

---

92. This case could arise if the security measures at issue will contribute positively to the general public good. *See, e.g.*, Keith N. Hylton, *When Should We Prefer Tort Law to Environmental Regulation?*, 41 WASHBURN L.J. 515, 517 (2002).

93. *See* Keith N. Hylton, *Optimal Law Enforcement and Victim Precaution*, 27 RAND J. ECON. 197, 198 (1996).

94. That means at least one element—q, L, or X—would need to be reduced.

95. *See generally* Gary S. Becker, *Crime and Punishment: An Economic Approach*, 76 J. POL. ECON. 169 (1968).

96. For example, a search of Factiva's database for software security or network security articles published in 2004 yielded 11,269 hits.

on the victim of an attack. Second, an attack can cause harm to business operations, including lost sales or lost advertising revenues while an e-commerce site is down. Although a company may see a general loss in sales, estimating the amount of damage is far more speculative than determining the tangible labor costs of bringing a network back online. Finally, an attack can cause harm to individuals by eroding their confidence in the safety of their personal information.[97] This kind of loss is difficult to quantify, but could be substantial. For example, in the ChoicePoint attack discussed above, many customers probably felt harmed by what they perceived as an invasion of their privacy.[98] Although the costs of the first type of damage are generally possible for the victim to quantify, those from the second type are virtually impossible to ascertain, as it would require valuation of intangible assets such as goodwill. As such, very little data currently exists that reliably tracks the economic effects of software system security breaches.

Available sources indicate that most corporations experience security breaches on a regular basis. One of the few sources of data on the economic costs of computer and network security breaches in the U.S. economy is the Computer Security Institute survey, conducted with the assistance of the San Francisco office of the Federal Bureau of Investigation. The CSI/FBI survey, which has been conducted annually since 1995, is sent to 5,000 information security practitioners within U.S. corporations, government agencies, financial institutions, medical institutions, and universities.[99] In the 2005 survey,[100] over half of the respondents reported "unauthorized use of computer systems within the last twelve months."[101] Although that percentage is relatively flat

97. This was a favorite subject among early e-commerce observers, who argued that a lack of faith in website security would limit on-line purchases, thus justifying government regulation. *See* Xiaorui Hu et al., *Myth or Reality: Effect of Trust-Promoting Seals in Electronic Markets*, *in* TRUST IN THE NETWORK ECONOMY 143, 146–49 (Otto Petrovic et al. ed., 2004), *available at* http://zlin.ba.ttu.edu/papers/Published/Trust-promoting.pdf (last visited Sept. 01, 2006) (asserting that consumer assessment of the safety of their personal information influences willingness to make online purchases).

98. *See* ACLU, FAQ on Choice Point, http://www.aclu.org/privacy/consumer/15301leg20050310.html (last modified Mar. 10, 2005).

99. CSI/FBI 2005 SURVEY, *supra* note 84, at 23.

100. The survey was distributed in January 2005 and reports incidents in 2004.

101. CSI/FBI 2005 SURVEY, *supra* note 84, at 11.

in comparison to both the 2003 and 2004 surveys, it is actually down from seventy percent in 2000.[102] However, as many as thirty-five percent of 2005 respondents reported a virus attack, and thirty-five percent reported "unauthorized access to information."[103]

Table 2: Computer Security Breaches and Damage Estimates

Type	Prevalence	Damages*
Virus Attack	35%*–68%†	$42.8 million
Unauthorized Access to Information	35%†–49%*	$31.2 million
Theft of Proprietary Information	~2%*–12%†	$30.9 million
Denial of Service	49%	$7.3 million
Total Damages		$130.1 million
Average Damages per Respondent		$204,000

Note: Damage amounts are based on 2005 data and are given in 2005 dollars. All amounts are rounded to the nearest $100,000.

Source: *2005 CSI/FBI Computer Crime & Security Survey; prevalence statistics based on 700 responses; damages statistics based on 639 responses.

† 2005 Forrester IT Security Threats Survey, based on 188 responses.

Other sources corroborate CSI/FBI's finding of widespread security breaches. For example, a January 2005 Forrester survey of 200 technology decision makers identified viruses, worms, and "[e]mployees acting in unauthorized ways" as substantial security threats.[104]

Although dollar losses associated with these attacks are fairly costly in the aggregate, they may be in decline. According to the 2004 CSI/FBI survey, security breach losses during the previous year totaled $141.5 million among respondents who could provide an estimate (269 respondents).[105] In 2005, estimated losses fell

102. *Id*.

103. *Id.* at 13.

104. *See* David Friedlander, Forrester Research, IT Security Threats in 2005: Viruses and Worms Top The List (Mar. 25, 2005), http://www.forrester.com/Research/Document/Excerpt/0,7211,36640,00.html.

105. LAWRENCE A. GORDON ET AL., 2004 CSI/FBI COMPUTER CRIME AND SECURITY SURVEY 10 (2004) [hereinafter CSI/FBI 2004 SURVEY].

to $130.1 million for those respondents who could provide a dollar amount (639 respondents).[106]

Average loss per respondent also fell from 2004 to 2005. In 2004, the average loss per respondent was $526,000;[107] by 2005 that figure had decreased to $204,000.[108] The CSI/FBI report attributes the declining losses to "increased awareness of, and improved technology to cope with some threat types, such as viruses."[109] In recent years, typical viruses "have spread far more slowly than their antidotes, at least where enterprise networks are concerned."[110]

The 2005 CSI/FBI separated losses by source of attack. The survey found that viruses cost respondents nearly $42.8 million; unauthorized access cost $31.2 million; and theft of proprietary information resulted in another $30.9 million in losses.[111] Denial of service attacks ran a distant fourth, creating $7.3 million of respondent losses.[112] Respondents employed a variety of technologies to combat these security threats. According to IDC, the most common respondent security technology was antivirus software, perhaps because virus attacks have traditionally caused the greatest dollar losses.[113] Firewall and anti-spyware software came in second and third.[114]

Although the damage figures appear modest when compared with a 2004 U.S. GDP of over $12 trillion,[115] it is important to note that the CSI/FBI survey may not provide an accurate estimate of overall damages. The survey respondents consisted of IT security personnel, typically highly technical professionals attuned to their organization's computer systems. Accordingly, the respondents were well-informed regarding the specific system costs resulting from attacks, but were poorly positioned to quantify the business

---

106. The 2002 survey reported the highest dollar loss of recent years: $455.8 million. *Compare* CSI/FBI 2005 SURVEY, *supra* note 84, at 14 *with* ROBERT RICHARDSON, 2003 CSI/FBI COMPUTER CRIME & SECURITY SURVEY 20 (2003) (listing the dollar losses associated with several recent years).

107. *See* CSI/FBI 2004 SURVEY, *supra* note 105, at 10.

108. *See* CSI/FBI 2005 SURVEY, *supra* note 84, at 14. All amounts have been rounded to the nearest $100,000.

109. *Id*.

110. *Id*.

111. *Id.* All amounts have been rounded to the nearest $100,000.

112. *See id.*

113. CSI/FBI 2005 SURVEY, *supra* note 84, at 14.

114. CHARLES J. KOLODGY ET AL., IDC, 2005 ENTERPRISE SECURITY SURVEY 16 (2005).

115. *See* CIA, The World Factbook—United States, http://www.cia.gov/cia/publications/factbook/geos/us.html (last visited Oct. 14, 2006).

or customer costs associated with the attacks. As the anecdotes relayed earlier indicate, the business costs of a security breach might be far greater than the system-specific costs.

Recent academic research attempts to estimate better the impact of computer security breaches on public corporations by using stock market event studies. For example, Katherine Campbell and her colleagues study the costs of security breaches by using the public announcement of breaches as events.[116] Examining stock market data from January 1995 through December 2000, their study finds "a highly significant negative market reaction for information security breaches involving unauthorized access to confidential data, but no significant reaction when the breach does not involve confidential information."[117] In other words, although viruses may receive much of the press attention and may affect more people in the aggregate, they do not appear to alter the stock market's valuation of a firm. However, unauthorized access to proprietary data or confidential consumer files does appear to influence stock prices.[118]

Anecdotal evidence corroborates the view that some computer attacks have an effect on the financial standing of companies. For example, when eBay's website was brought down by hackers in 2000, the firm's CEO reported that the economic impact was "[m]inimal . . . . Even if you make extreme assumptions that none of [the] deferred listings ever got made, the maximum we could be talking about is $50,000 of lost revenue to eBay."[119]

These statistics recording the costs of cyber attacks should generally be viewed with some skepticism. First, low response rates increase the variance of results and may magnify any sampling problems. Accordingly, surveys may over or understate the true costs of security breaches depending on unknown bias. For example, the CSI/FBI survey was sent to 5,000 entities in 2005, but only 700 responded.[120] If survey response is correlated with breach losses, the CSI/FBI figures could overstate the population

---

116. Katherine Campbell et al., *The Economic Cost of Publicly Announced Information Security Breaches: Empirical Evidence from the Stock Market*, 11 J. COMPUTER SECURITY 431, 431–32 (2003).

117. *Id*. at 431.

118. The Federal Trade Commission estimated that data theft caused $50 billion in losses in the U.S. in 2004. *Hot Data*, ECONOMIST, June 25, 2005, at 14–15.

119. George Anders, *EBay's CEO Reacts to Hacker Attack, Suggests Joint Action on Web Security*, WALL ST. J., Feb. 10, 2000, at B18.

120. *See* CSI/FBI 2005 SURVEY, *supra* note 84, at 2. In earlier surveys, the response rate was lower because CSI issued fewer reminders.

average. According to the CSI/FBI 2005 survey, however, only twenty percent of respondents report computer security crimes to the authorities.[121] An overriding concern about negative publicity appears to be the main reason behind corporate silence on security vulnerabilities.[122] Accordingly, those entities hit hardest by security breaches may choose not to report, and the estimates may actually understate the damages incurred.

It is also significant to distinguish the private damage estimates for a firm or individual from the greater social costs. Consider an online bookseller that is taken offline by an attack but comes back online in an hour. What portion of expected sales did it lose? Consumers may have purchased their books from another unaffected online seller or they may have purchased them through their local bookstore instead of using the internet, both of which mean the seller that suffered an outage lost all expected sales for the hour it was down. Alternatively, consumers may have simply waited until their online seller of choice was back up, so the bookseller did not lose any sales from the outage. Sorting through these factors implies that the estimates thrown out in early news reports can overstate security breach costs by a considerable amount.[123] More fundamentally, in each of the bookstore scenarios the same goods are eventually sold to consumers. It is "just" a matter of when the books are sold and who the seller is. Thus, the U.S. economy suffers no lost sales cost at all under the above example, even though the online bookseller whose site goes down does.[124] Only if consumers forgo purchases altogether, perhaps because the need is time sensitive, would the cost of "lost sales" be added to the social ledger. It is also difficult to discern the variation in costs. Few surveys track these costs, and virus and worm damages are always reported together—some worms simply slow down networks, although others may cause significant damage by destroying files.

---

121. *Id.* at 19.

122. *See id*.

123. For example, in 1988, the *Los Angeles Times* estimated that one of the first worms caused damages of approximately $97 million. *Damage From Computer Virus Set at $97 Million*, L.A. TIMES, Nov. 18, 1988, § 4, at 4. But during the coder's trial, the federal district court awarded actual damages of only $150,000, with individual websites suffering damages ranging from $200 to $53,000. *See* Michael P. Dierks, *Computer Network Abuse*, 6 HARV. J.L. & TECH. 307, 317 (1993).

124. Of course, that bookseller also incurs the direct cost of restoring the network.

The review of quantitative damages associated with attacks provides a more nuanced picture than one would surmise from reading newspaper accounts. On the basis of available data, one could conclude that website defacements and some viruses and worms are closer to nuisances than catastrophes. Recall that one of the best-known worms in recent years, Blaster, did little more than slow down personal computers. On the other hand, the Love Bug overwrote and corrupted data files, causing some $10 billion in damages. Hence the only clear conclusion derived from the data is that the effects of software system security attacks are every bit as varied as the methods of attack.

2. Measuring Prevention Efforts

Another way to evaluate the economic impact of computer security threats is to consider how much effort end users expend to avoid them. As noted above, the amount an individual or firm is willing to spend to avoid a problem can be interpreted as a lower bound for the damage the problem might be expected to cause. In Table 3, we examine the available data on defensive steps taken by software users—government agencies, companies, and consumers—to reduce the impact and likelihood of attacks.

As measured by the number of companies taking preventative actions, security problems are even more widespread than indicated by the damage figures. The CSI/FBI surveys discussed above provide some useful information on the costs of prevention. The same caveat we discussed on damage estimates, regarding IT security specialists' view of damages, applies to these figures.[125] According to the 2005 survey, 97% of the companies and government agencies responding indicated that they use firewalls to prevent unauthorized access to their private networks.[126] Ninety-six percent reported using anti-virus software, and 72% indicated that they employ intrusion detection systems, which attempt to "identify and block malicious network activity in real time."[127] As many as 87% of respondents also conduct annual security audits.[128] These measures are not inexpensive. According to IDC, the IT security market was valued at $27 billion in 2004—a twenty percent increase over the previous year. Of that $27 bil-

125. *See infra* Part II.B.1.
126. CSI/FBI 2005 SURVEY, *supra* note 84, at 16.
127. *Id*.
128. *Id*. at 17.

lion, security services amounted to $12 billion, security software amounted to $10 billion, and security hardware amounted to $5 billion.[129]

Table 3: The Use of Security Measures at Corporations and Their Estimated Cost

Security Measures	% Employing	Cost of Security Measure
Firewall	58%† - 100%~	$911.9 million §
Anti-spyware Software	65%†	$29 million §
Anti-virus Software	43%† - 96%*	$2.7 billion §
Strong/Advanced Authentication	45%†	$599 million §
Intrusion Detection	31%† - 72%*	$366 million §
Annual Security Audits	87%*	
Total IT Budget Spent on Security	6-8%+	
Average Expenditure, Small Organization	19.9%^ of IT budget	$132,000^
Average Expenditure, Medium Organization	10.7%^ of IT budget	$360,000^
Average Expenditure, Large Organization	5%^ of IT budget	$1 million^
Average Expenditure, Very Large Organization	5.5%^ of IT budget	$6 million^

Notes: Dollar figures rounded to nearest $1,000 or million. Cost of security measures are assumed to be in year dollars for the year in which survey was conducted.

Sources: * signifies CSI/FBI (2005), *supra* note 83, + signifies Deloitte Touche Tohmatsu (2003), note 131, ^ signifies Briney & Price (2002), note 132, ~ signifies Ernst & Young, note 130, † signifies Forrester (2005), note 104, and § signifies IDC (2003 worldwide data), notes 86, 142, 178.

129. BURKE ET AL., *supra* note 26, at 1.

Other surveys find an equally high deployment of security tools among software users. For example, Ernst & Young's Global Information Security Survey reports that anti-virus desktop and server software is used by nearly 100% of respondent businesses.[130]

Additionally, respondents generally devoted a large share of their information technology budgets to security. According to a Deloitte Touche Tohmatsu survey conducted in 2003, IT security typically receives 6–8% of overall IT budgets in developed countries.[131] The 2002 *Information Security Magazine* survey reports that the percentage of IT budgets devoted to security ranges from 5% for large enterprises to 20% for small enterprises.[132] Government and financial services respondents in the *Information Security Magazine* survey were closer to the middle of that range at 8–9%.[133] These percentages translate into average budgets of $132,000 per year for small organizations (defined as those with 10 to 100 machines) and up to $6 million per year for very large organizations (those with 10,000 machines or more).[134]

Information from Table 2 and Table 3 can be used to help bound the expected losses incurred by businesses that do not invest in security. Equation (1) states that a user will invest in preventative measures (spending X) when those investments reduce expected costs (losses plus administrative costs) by

---

130. ERNST & YOUNG, GLOBAL INFORMATION SECURITY SURVEY 2004, at 23 (2004).

131. DELOITTE TOUCHE TOHMATSU, 2003 GLOBAL SECURITY SURVEY 11 (2003).

132. Andrew Briney & Frank Price, *Does Size Matter?*, INFO. SECURITY, Sep. 2002, 36, 38 tbl.C, *available at* http://infosecuritymag.techtarget.com/2002/sep/2002survey.pdf. One reason for the higher percentage among smaller firms appears to be fixed costs. Certain minimal expenditures are evidently required to protect information security. Smaller firms have lower overall IT budgets, and thus a larger share is devoted to security. *Id.*

133. *Id.* at 39 fig.3.

134. *Id.* at 40, 52. Other sources indicate state government spending on security is lower than federal spending: "But, generally, it is thought that the average amount of resources dedicated to IT systems is about 4 percent to 6 percent of a state's budget. The amount of money spent on securing those systems is a fraction of that percent." Garry Boulard & Janna Goodwin, *Cyber Terrorism: No Longer Fiction*, STATE LEGISLATURES, May 2003, at 22 (quoting Larry Kettlewell, chief information security officer for Kansas and a member of the National Association of State Chief Information Officers (NASCIO) security and liability team). To put these figures into context, consider advertising expenditures, which all businesses have. Dell, for example, spent $576 million, $473 million, and $426 million on advertising in 2005, 2004, and 2003, respectively. Dell Inc., Annual Report (Form 10-K), at 42 (Feb. 25, 2005).

enough to offset the investment: $pL + pC > X + qL + qC$. Table 2 provides average damage estimates of about $204,000 per respondent.[135] This provides a crude measure of *qL*, the expected losses from a security breach. Similarly, Table 3 provides a rough estimate of *X*, the cost of investing in security, of about $132,000 for a small organization, $360,000 for a medium organization, $1 million for a large organization, and $6 million for a very large organization. Accordingly, *qL*+*X* might measure approximately $340,000 for a small organization, $560,000 for a medium organization, $1.2 million for a large organization, and $6.2 million for a very large organization. Using census data on the number of firms,[136] and assuming that each person employed by a firm uses one computer,[137] gives a lower bound for the expected losses associated with businesses not investing in security of approximately $380 billion. That is, if businesses chose not to invest in security, the social losses would measure at least $400 billion. Roughly $340 billion of these total losses are associated with small organizations.[138]

Individual end users also purchase security products. For example, corporations devote resources to security awareness training for their employees.[139] Several security tool providers,

---

135. Average damages per respondent in 2005 is adjusted to 2002 dollar value by using a deflator of approximately 0.921, which is derived from the CPI averages of those two years. *See* Bureau of Labor Statistics, Consumer Price Index 1913–2006, ftp://ftp.bls.gov/pub/special.requests/cpi/cpiai.txt (last visited Nov. 19, 2006).

136. The U.S. Census Bureau defines a firm as "a business organization consisting of one or more domestic establishments in the same state and industry that were specified under common ownership or control." U.S. Census Bureau, Statistics of U.S. Business, Explanation of Terms, http://www.census.gov/epcd/susb/susbdefs.htm#firm (last visited Oct. 14, 2006). The size of an organization is defined as follows: small organizations have between 10-100 employees and total approximately 1.1 million firms in 2002; medium organizations have between 100 and 1,000 employees and total 90,000 firms in 2002; large organizations have between 1,000 and 10,000 employees and total 7,500 firms in 2002; and very large organizations have over 10,000 employees and total approximately 900 firms in 2002. U.S. Census Bureau, Statistics About Business Size, http://www.census.gov/epcd/www/smallbus.html#EmpSize (last visited Oct. 14, 2006).

137. Calculations are estimated in 2002 under the assumption that each person in an organization has a computer. These calculations are meant to be illustrative given the nature of the assumptions. However, even if the expected losses of not investing in security for small organizations is cut in half, there would be expected losses of over $175 billion from not investing in security.

138. Note that *p*, *L*, and *C* can vary across firms, and a similar methodology can be applied.

139. CSI/FBI 2005 SURVEY, *supra* note 84, at 1.

such as McAfee, Symantec, Panda Software, and MicroWorld Technologies, sell tools aimed at individual consumers.[140] Further, the consumer market is large; consumer security software sales produced several billion dollars of revenue in 2004.[141] The total worldwide consumer antivirus software market totaled $821.3 million in 2003, nearly 25% higher than 2002.[142] Moreover, consumers are either demanding security features in their hardware purchases, or are at least willing to pay more for them.[143] For example, many personal computers now come with preloaded security feature options.[144]

Consistent with an increased demand for more secure software, software developers have also begun to invest in security-focused development procedures.[145] For instance, Microsoft halted all feature development on Windows products for two months in 2002 so that the entire development team (of approximately 8,500 people) could focus on ways to build more

---

140. Symantec leads the market for consumer antivirus software with 88% of the market, while McAfee holds 11.8%. Bill Snyder, McAfee Hacks Into Success (June 27, 2005), http://www.thestreet.com/pf/tech/billsnyder/10229668.html.

141. Symantec generated revenues of $1.87 billion and McAfee generated revenues of $911 million in 2004. SYMANTEC CORP: SYMANTEC 2004 ANNUAL REPORT (2004), *available at* http://media.corporate-ir.net/media_files/irol/89/89422/FileUpload/Symantec_2004_AnnualReport.pdf; MCAFEE INC., AT-A-GLANCE 1 (2006), *available at* http://www.mcafee.com/us/local_content/media/fc_mcafee_at_a_glance.pdf.

142. BRIAN E. BURKE, IDC, WORLDWIDE ANTIVIRUS 2004-2008 FORECAST AND 2003 VENDOR SHARES 6 (2004).

143. As another reflection of the value that consumers place on security, several recent consumer-oriented firm acquisitions have been consummated at a premium over market value. For example, Symantec bought data management company Veritas for $13.5 billion in 2005; the deal valued Veritas stock at $30.78, which was a 9.5 percent premium over closing price during merger. *See* Symantic Inc., Acquisitions, http://www.symantec.com/about/profile/development/acquisitions/index.jsp. Computer Associates' purchase of Netegrity (an identity management software company) in 2004 was valued at $430 million, or $10.75 per common share. This deal reflects a 39% premium on the day before the deal was announced. Robin Arnfield, *Computer Associates Buys Netegrity*, CIO TODAY, Oct. 8, 2004, http://www.cio-today.com/story.xhtml?story_id=27466.

144. *See*, *e.g.*, Jim Finkle, *Dell to Preload Trend Micro Software on PCs*, PC MAG., Oct. 28, 2005, *available at* http://in.tech.yahoo.com/051028/137/60s18.html (discussing Dell's announcement that it would preload antivirus software).

145. Further support can be found in the finance community. Venture capitalists have begun to show interest in firms developing automated security testing tools. *See, e.g.*, Vauhina Vara, *Tech Companies Check Software Earlier for Flaws*, WALL ST. J., May 4, 2006, at B1; Stacey Higginbotham, Software Security Attracts VCs, (Jan. 31, 2006), http://www.thedeal.com/servlet/ContentServer?pagename=TheDeal/TDDArticle&bn=NULL&c=TDDArticle&cid=1106611797067; Howard A. Schmidt, *Give Developers Secure-Coding Ammo*, CNET NEWS.COM, Nov. 3, 2005, http://news.com.com/Give+developers+secure-coding+ammo/2010-1002_3-5929364.html.

secure software.[146] This effort is estimated to have cost the company at least $200 million in lost productivity.[147] The company now has a required training program and a software development process that tries to address potential security concerns throughout all stages of software development.[148] Other software developers are in the early stages of instituting similar security measures.[149] These firms are presumably focusing more effort on security because they believe it will contribute to their long-term bottom lines.

In summary, the data suggest that security is important for both companies and consumers. Businesses devote significant portions of their IT budgets to protecting their software systems from attack. Consumers, who are relatively uninformed of security threats compared with corporations, appear to value security enough to support a sizeable software and hardware industry devoted to protecting home systems. Software developers recognize these concerns and are beginning to respond by employing security-oriented development procedures.

C. The Underlying Market Failures

The estimates presented above suggest that software system security is a serious concern. The costs imposed by some security breaches can be significant. Companies and individuals are willing to spend considerable amounts to prevent security breaches, and security prevention expenditures have increased over time.[150] With problems of this nature, the market should

---

146. *See* Michael Howard & Steve Lipner, *Inside the Windows Security Push*, IEEE SECURITY & PRIVACY MAG., Jan.–Feb. 2003, at 57.

147. *See* Robert Lemos, *Gates Vows Better Security for Customers*, CNET NEWS.COM, Jan. 24, 2003, http://news.com.com/2100-1001-981955.html.

148. Procedures such as "threat modeling" and "risk assessment" are incorporated in all phases of development, in addition to the more traditional "penetration" testing that is done once the code is written. Secure engineering procedures are explained in Gary McGraw, *Software Security*, IEEE SECURITY AND PRIVACY, March–April 2004, at 80.

149. *See* Paula Rooney, Is Windows Safer?, CRN.com, (Feb 10, 2006), http://www.crn.com/sections/coverstory/coverstory.jhtml;jsessionid=VV1Q351RM5A1YQSNDBOCKH0CJUMEKJVN?articleId=179103240 (observing that "rivals Sun Microsystems, Novell and Red Hat plan to detail their own security enhancements for Solaris and Linux.").

150. IDC figures document the growth of the worldwide threat management security appliance market, which includes firewall and virtual private networks, intrusion detection and prevention, and unified threat management measures. The market grew from $1.8 billion in 2003 to $2.5 billion in 2004, which represents

respond with solutions. If solutions do not readily emerge, one would generally conclude that the market has failed in some way.

1. Key Market Failures

Two primary market failures have been suggested in the provision of software security. First, there may be differences in the amount of information readily available to different parties, resulting in "information asymmetries" that could lead to an inefficient provision of security. Second, individual end users may undervalue security in relation to the optimal level from society's point of view. We address each of these in turn.

In the textbook case of information asymmetries, one party (typically the seller) knows the quality of the good to be sold whereas the other party (the buyer) knows little or nothing about the quality of the good. The buyer cannot trust the seller's assurances of quality because the seller has an incentive to overstate quality in order to make a sale. Unless buyers can find some useful signal of quality, they will be unable to distinguish easily between low-quality and high-quality goods. The result is that low-quality goods can often push high-quality goods out of the market. This dilemma is illustrated by the well-known "lemons market" problem, with used cars serving as the standard example. The problem asserts that only "lemons" will be sold in the used car market unless someone figures out a way to prove to the buyer that the car is not a lemon.[151]

Some legal scholars argue that software security is an example of a lemons market.[152] One argument suggests that, because end users cannot examine complicated source code to assess its security vulnerabilities, they are at the mercy of suppliers.[153]

---

a percentage increase of about roughly 38%. By 2009, IDC estimates that the market will reach roughly $5 billion. CHARLES J. KOLODGY, IDC, WORLDWIDE THREAT MANAGEMENT SECURITY APPLIANCES 2005–2009 FORECAST AND 2004 VENDOR SHARES: SECURITY APPLIANCES REMAIN A WELL-OILED MACHINE 11–12 (2005).

151. *See* George A. Akerlof, *The Market for "Lemons": Quality Uncertainty and the Market Mechanism*, 84 Q.J. ECON. 488, 489–90 (1970).

152. *See, e.g.*, Barnes, *supra* note 7, at 292–93.

153. *See, e.g.*, *id*. at 292. ("As long as software is maintained as a trade secret, and development occurs behind closed doors, buyers have nothing more to go on than vague, unprovable assertions about quality and security (which are cheap to make)."). The belief that opening code will solve any asymmetric information problems is likely misguided. Only those users specially trained in software secu-

Suppliers will have little incentive to add high levels of security because the buyer has no low-cost method for ascertaining quality. Thus, the security level provided by developers is consistently lower than it should be.

The traditional lemons market argument, however, does not strictly apply to software without some important modifications. A given software program, unlike a given make or model of used car, does not vary in "quality." Each copy is an exact replica of the original. Although the environment in which the program is used varies considerably between users, one software copy will not be defective in the same sense that one used car may be defective.

Nonetheless, the lemons argument could apply to different types of software, as opposed to individual copies of a given program. For example, if one word processor program is more secure than another, but consumers cannot easily observe the security difference, then consumers may not be willing to pay more for the more secure program and developers have no incentive to invest in security. The end result could be that security is reduced below some optimal level. Of course, this need not be the case if there are ways to signal the true quality of a particular program.[154]

Proponents of the software lemons market view argue that objective signals of the quality of software security are lacking. Firms specializing in security systems and tools do exist, but these firms have incentives to exaggerate security dangers in order to sell their security services.[155] Although large enter-

---

rity would be able to identify potential problems within the source code. Even if users rely on specialists, finding a problem can be like searching for a needle in a haystack. The well-known buffer overflow problem mentioned earlier is only four short lines of seemingly innocuous code, which would be buried in the hundreds of lines of code comprising a program. *See* Sandeep Grover, *Buffer Overflow Attacks and Their Countermeasures*, LINUX J., Mar. 10, 2003, http://www.linuxjournal.com/article/6701. Reinforcing this point, vulnerabilities are routinely found and exploited in open source software, just as they are in closed source, proprietary software. *See supra* note 56. This is not to say that the lemons market argument does not hold, but opening source code for general review would not solve the problem.

154. For instance, reported expenditures on investments in secure development procedures and training by software makers can signal commitment to improved security engineering.

155. Skibell, *supra* note 75, at 131 ("A number of companies search for software vulnerabilities and publicly report them. The primary motivation for these companies is to generate publicity for their security services, and they have an economic incentive to issue a large amount of these alerts.").

prises may be able to conduct software security reviews, or hire the task out to specialists, small businesses and individual consumers are far less likely to evaluate security issues on their own. Furthermore, small businesses and consumers are more likely to install software improperly, negating built-in security features.

Contracts could possibly offer a workable solution to the lemons market problem, but most scholars feel they have not done so thus far.[156] Software users could insist on contract terms that place at least some of the security burden on a provider—for instance, by requiring a product refund from a software developer if security problems arose. Although terms of this kind may be a feature of large, custom software negotiations, they are not found in off the shelf software. One of the primary problems is identifying an appropriate party for contracting.

Software production and use involves a long list of participants, any one of which might be held legally responsible for security. Consider the path a program takes from inception to use. A group of programmers within a company,[157] typically each with their own areas of expertise, collectively plan, write, and test a program.[158] It is then licensed, say to another business, whose IT staff installs the program on their company network. Employees within the company then use the program. Now add the internet to the picture, and one must factor in backbone network providers, internet service providers, and hackers searching the internet for system weaknesses. Improper program installation or maintenance by corporate IT staff, careless password protection by end users, and improper security procedures by internet access providers may provide

---

156. Lichtman and Posner state the rule succinctly: "[W]here transaction costs are low and employees have adequate resources, contracts allow private parties to shift and divide legal responsibility as they see fit." Doug Lichtman & Eric Posner, *Holding Internet Service Providers Accountable* 10 (John M. Olin Law & Econ. Working Paper No. 217, (2d ser.) 2004), *available at* www.law.uchicago.edu/Lawecon/WkngPprs_201-25/217-dgl-eap-isp.pdf. Here, the transaction costs are not low, as we describe, since multiple parties are involved.

157. This analysis considers for-profit, closed source software, but an analogous story could be told for software developed under the open source model.

158. One scholar gives the example of an account software package, which requires (at a minimum) a programmer and an accountant. "Each is an expert in his own field, but neither is really an expert relative to the final product." Michael C. Gemignani, *Product Liability and Software*, 8 RUTGERS COMPUTER & TECH. L.J. 173, 190 (1981).

as much risk of security breach as programming vulnerabilities. If a user successfully negotiated a contract that included security terms with a software developer, security weaknesses in the other links of the chain could still expose the system to attacks.[159] Moreover, the presence of a contract with a software developer could induce other parties—internet service providers or the user herself—to take fewer security precautions, which could further raise the risk of a security breach.

The second market failure concerns business and consumer users undervaluing security. Even if end-users could easily discern the quality of software security, they still would not value it as highly as they should from a societal perspective. One illustration of the argument is as follows: by securing networks himself, a user closes off one entry route for would-be hackers, benefiting himself and others on the larger, interconnected network.[160] In deciding on the level of security for the system, though, the user does not consider the benefits accruing to anyone but himself.[161] As with information asymmetries, this is a well-known problem in economics called externalities.[162] There are many ways of handling these externalities, ranging from taxing the activity of concern to introducing new laws.[163]

In some cases, however, policies can actually accentuate security externalities. For example, fraud policies and legislation frequently exacerbate the free-rider problem among consumers. If a consumer's credit card number is stolen and used for fraudulent purchases, the individual cardholder is not respon-

---

159. Lichtman and Posner raise a related problem with contracts over software security: "[A]ny network of contracts focusing on issues of cyber-security would be perpetually out of date, and updating such a complicated web of interdependent security obligations would be all but impossible given the number of parties involved and the complicated questions any update would raise regarding the appropriate adjustments to the flow of payments." Lichtman & Posner, *supra* note 156, at 17.

160. Picker argues that the development of remote control, such as zombies and "bots" exacerbates the problem: "This has made the decentralized decisions of end-users much more salient. My failure to manage my computer appropriately puts you at risk." Picker, *supra* note 5, at 3.

161. *See generally* Steven Shavell, *Individual Precautions to Prevent Theft: Private Versus Socially Optimal Behavior*, 11 INT'L REV. L. & ECON. 123 (1991) (setting forth a general model that addresses this issue).

162. *See generally* ARTHUR CECIL PIGOU, THE ECONOMICS OF WELFARE (1924).

163. *See, e.g.*, WILLIAM J. BAUMOL & WALLACE E. OATES, THE THEORY OF ENVIRONMENTAL POLICY 155–210 (2d ed., Cambridge Univ. Press 1988) (1975).

sible for any of the resulting losses.[164] Either the store selling the goods to the imposter assumes the loss because proper security practices were not followed, or the bank issuing the credit card assumes the loss. Although rules of this sort appeal to a sense of justice and equity, they also reduce an individual's incentives to take security precautions.

In addition to causing underinvestment in prevention, the fact that online security involves externalities has implications for after-the-fact security fixes as well. For instance, end users and corporations may not install software "patches"—code distributed by the original software developer intended to repair security vulnerabilities—within an optimal timeframe.[165] Indeed, the Computer Emergency Response Team (CERT) at Carnegie Mellon University, which tracks computer security incidents, has found that most intrusions exploit known vulnerabilities for which counter-measures are readily available.[166] Thus, patches are available, but end users do not take advantage of them. Further, there are many cases where consumers and workers are unaware that the problem and its solution even exist. As the CSI/FBI survey observes, "For some time, it has been widely recognized that computer security is as much a management problem as it is a technology problem."[167] Even for corporations with a dedicated IT department, individual corporate users often must individually download or accept a patch in order to apply it to their personal computer. Hackers know this, and therefore some continue to use existing hacking tools even as the more skilled among them search for new ones in light of the patch.[168]

---

164. *See, e.g.*, Visa, Visa Security Program, http://usa.visa.com/personal/security/visa_security_program/zero_liability.html (last visited Nov. 20, 2006).

165. Moreover, since IT employees at corporations and in government agencies act as agents for their employers, moral hazard problems likely exacerbate the underinvestment problem. That is, a company may value security less than is socially optimal, but unless it creates clear incentives for its employees, those responsible for enacting the company's security policy may underinvest even from the company's perspective.

166. Countermeasures here are loosely defined, covering everything from proper password procedures to installing patches. *See* CERT COORDINATION CTR., CARNEGIE MELLON SOFTWARE ENG'G INST., CERT/CC OVERVIEW: INCIDENT AND VULNERABILITY TRENDS 14 (2003), *available at* http://www.cert.org/present/cert-overview-trends/module-2.pdf.

167. CSI/FBI 2005 SURVEY, *supra* note 84, at 17.

168. One press article described the scenario as follows: once a patch is released, "a curious race begins. Digital vandals—those who write worms, viruses and other rogue programs—eagerly download the patch and reverse-engineer, taking it apart to search for clues on how to exploit the very . . . security hole the patch

Other market pressures can exacerbate problems in the provision of software security. The one quality aspect readily apparent to all end users, both corporate and consumer, is the number and function of features available in the software. Features are what sell software, but adding more features can reduce security. As some security experts have observed, "Complexity is the worst enemy of security, and systems that are loaded with features, capabilities, and options are much less secure than simple systems that do a few things reliably."[169]

The Trusted Computer System Evaluation Criteria, launched by the U.S. Department of Defense in 1985, illustrates the tradeoff between the level of security and the number of program features.[170] The program, dubbed the Orange Book, attempted to establish a general method for government agencies to evaluate security requirements. The Orange Book provided software developers with specific criteria that could be used to measure the security of their programs. Federal agencies were then supposed to use the same criteria in evaluating the security of the software they purchased for internal uses. By all accounts, the program was not a success. According to a document prepared by the National Research Council's Computer Science and Telecommunications Board:

> Customers buy features and performance rather than security. The failure of the U.S. government's Orange Book program even within the federal marketplace is a striking example: the government demanded secure systems, industry produced them, and then government agencies refused to buy them because they were slower and less functional than other non-secure systems available on the open market.[171]

The arguments detailing both of these potential market failures in software system security—asymmetric information and externalities—are quite persuasive; however, a closer examination leads us to question their importance for all parties.

---

was meant to cover." Steve Lohr, *Fixing Flaws, Microsoft Invites Attack*, N.Y. TIMES, Sept. 29, 2003, at C1.

169. Bruce Schneier & Adam Shostack, SECURITYFOCUS.COM, Results, Not Resolutions (Jan. 24, 2002), http://www.securityfocus.com/news/315.

170. Dep't of Def., TRUSTED COMPUTER SYSTEM EVALUATION CRITERIA (1985), *available at* http://csrc.nist.gov/secpubs/rainbow/std001.txt.

171. COMPUTER SCI. & TELECOMM. BD., NAT. RES. COUNCIL, CYBERSECURITY TODAY AND TOMORROW: PAY NOW OR PAY LATER 9 (2002), *available at* http://csrc.nist.gov/publications/reports/CSTBNRC-report.pdf.

2. *Are the Market Failures Significant?*

First consider the market failure that could arise from information asymmetries. The lemons market was a genuine problem for used automobiles at a certain point in time. Now, however, a number of non-regulatory measures help to solve the problem of information asymmetry. For example, used car dealers now offer their own warranties and original car manufacturers provide "certified-pre-owned" cars for leasing and purchasing, both of which provide meaningful signals about the quality of the cars.[172] Moreover, a third-party information system, Carfax, has emerged, providing background checks for used cars.[173] In fact, private sector information providers and product evaluators are common in a number of industries throughout the economy, from product safety testing with Underwriters Laboratory to financial stability ratings for bond issuers with Moody's or Standard & Poor's.[174] These various market initiatives have led some software security experts to ask, "Can we take useful lessons from [the developments in used cars] for security?"[175]

One such lesson is that patience may be the best remedy for at least some users and some products. Market derived signals of software system security are generally superior to government derived signals because they are more flexible and more likely to match cost with value. Moreover, it does not appear that the wait for business-oriented market solutions will be prohibitively long. Some third-party evaluators for enterprise software users are already emerging, albeit for narrow applications. For example, BITS Financial Services Security Lab tests and certifies hardware and software for online banking and

---

172. *See* Lexus.com, Lexus Certified Pre-Owned Overview, http://www.lexus.com/cpo/overview/index.html (last visited Dec. 7, 2006) (exemplifying a certified pre-owned guarantee that ensures quality inspections and reconditioning in addition to a three year warranty); *see also* Sanford Grossman, *The Informational Role of Warranties and Private Disclosure about Product Quality*, 24 J. L. & ECON. 461, 470 (1981) (stating that Akerlof's findings do not hold after one considers manufacturer warranties).

173. Carfax, http://www.carfax.com (last visited Oct. 2, 2006).

174. *See* Harold Furchtgott-Roth, Robert W. Hahn & Anne Layne-Farrar, *The Law and Economics of Regulating Ratings Firms*, 3 J. COMPETITION L. & ECON. (forthcoming 2007).

175. ADAM SHOSTACK, AVOIDING LIABILITY: AN ALTERNATIVE ROUTE TO MORE SECURE PRODUCTS 2 (Mar. 5, 2005), http://www.homeport.org/~adam/beyondliability.pdf.

related financial services.[176] Such firms do not provide security solutions or tools and thus do not have conflicts of interest in rating the security of a developer's programs. As a result, their security ratings can be viewed as objective signals that can reduce asymmetry of information.

Software security evaluation services are also available for small businesses and consumers. These services frequently rate software products according to a scale developed by Common Criteria.[177] Common Criteria's rating scale can also help those software manufacturers with high ratings gain acceptance among customers that value security. The rating thus provides a "seal of approval" for security.

Individual consumers and small firms, however, must be aware of possible security problems before they are likely to take advantage of emerging rating services like Common Criteria. Most consumers and small business users are unable to assess the security of a software product on their own. Industry reports, however, reveal an increase in the availability of burgeoning security products aimed at these two segments—ranging from anti-spyware software products to plug-and-play firewall appliances—and suggest that things are slowly changing.[178] IDC notes that "[c]onsumers are slowly understanding that any machine connected to the internet is a potential target for hackers or automated attacks."[179]

The growing number of press articles on security issues should increase awareness among consumers and small businesses. The more consumers and businesses read about security breaches, the more knowledgeable they will become, making it possible that a reputational mechanism could discipline

---

176. BITS Product Certification Program, http://www.bitsinfo.org/c_certification.html (last visited Nov. 20, 2006). The company is off to a slow start, however, certifying only one product in its first three years of operation.

177. Common Criteria is overseen by the National Information Assurance Partnership, a "U.S. Government initiative between the National Institute of Standards and Technology (NIST) and the National Security Agency." National Information Assurance Partnership, http://niap.nist.gov/cc-scheme/index.html (last visited Nov. 7, 2006).

178. *See, e.g.*, CHARLES J. KOLODGY, IDC, WORLDWIDE FIREWALL SOFTWARE 2004-2008 FORECAST AND 2003 VENDOR SHARES: DESKTOP FIREWALLS ON THE MOVE (2004); BURKE, *supra* note 142; CHARLES J. KOLODGY, IDC, WORLDWIDE INTRUSION DETECTION AND PREVENTION 2004–2008 FORECAST AND 2003 VENDOR SHARES: INTRODUCING THE FIREDOOR (2004).

179. *Id.* at 6.

companies with the worst security records.[180] A combination of reputation and third-party evaluators could solve the market failure, at least for some market segments. Highly visible companies, such as Microsoft, Novell, Oracle, and Sun, have all either vowed to improve the security of their software, or have already begun to do so—suggesting that the reputation costs of negative press reports are significant enough to justify investment in improved security. We conclude, therefore, that the first market failure—asymmetric information—is currently a serious problem only for some software types and only for consumers and small businesses. And, even for these groups, signs of improvement are evident.

The second market failure, externalities associated with not taking adequate security precautions, may not be as significant as first suggested for either businesses or consumers. Consider spillovers from individual users first. Suppose two people, A and B, are both connected to the internet. Person A frequents chat rooms and downloads files indiscriminately. As a result, person A is a probable candidate for viruses, and may even have her computer taken over surreptitiously as a zombie. Person B, in contrast, has a firewall, has installed anti-spyware and antivirus software, and accepts automatic updates to that software sent by the software company. She never visits chat rooms on the internet, and is careful to download files only from trusted sources. Given her precautions, is person B likely to be infected simply because person A is? Probably not. Although person A may assist in spreading a virus, person B either will not receive it because her firewall will block it, or will not open it because she exercises caution in opening files or because her automated antivirus software will rapidly fix the problem. If, however, person A's computer is taken over as a zombie that is then used by a malicious hacker for a distributed denial of service attack, person B may suffer because one of her favorite websites is taken down. Additionally, the attacked

180. Microsoft's increased emphasis on security in development could be viewed as a reaction to critical press on the security of the company's products. Microsoft emphasizes security in press releases, suggesting that reputation effects are, or are becoming, important. Press Release, Microsoft, Gates Highlights Progress on Security, Outlines Next Steps for Continued Innovation (Feb. 15, 2005), http://www.microsoft.com/presspass/press/2005/feb05/02-15RSA05KeynotePR.mspx; Press Release, Microsoft, Gates Shares Microsoft's Vision for a More Secure Future (Feb. 14, 2006), http://www.microsoft.com/presspass/press/2006/feb06/02-14RSA06KeynotePR.mspx.

website may also suffer economic harm unless it has taken sufficient precautions to prevent denial of service attacks.[181]

Despite the potential costs, even the relatively well-informed who, like person B, are aware of security issues, may rationally choose not to take the precaution of installing every patch for software bugs. As noted earlier, CERT found that most security breaches exploit weaknesses for which patches already exist. Some have interpreted this finding as evidence that individuals undervalue security.[182] The observed behavior instead might reflect a rational decision on the part of end users or system administrators in the face of prevention costs that exceed the benefits of reduced damages. CERT identified a total of 5,500 security vulnerabilities in 2002.[183] Suppose that a large company's computer system were susceptible to only 5%, or 225 of those vulnerabilities, and that the IT department knew perfectly which 5% that was. Further, suppose that each vulnerability announced took 15 minutes to read and understand, and each patch took an hour to download and install correctly within the company's system. Devoting 8 hours a day solely to security patching, it would take an IT administrator approximately 9 days to read about all the vulnerabilities that affected the company's computer system and another 34 days to download and install the patches. Large corporations fully aware of the vulnerabilities and the patches to fix them might still decide to focus only on the truly serious threats given such high prevention costs. Especially for nuisance category vulnerabilities, the cost of fixing the security problem might far exceed the benefits. Moreover, patching costs could be even higher for small businesses and individual consumers, who lack the resources an IT department can provide.

Many firms, however, do appear to have significant incentives to protect their systems. Companies with truly vital systems generally "air gap" them—meaning that those systems are not linked directly to public networks and are therefore not

181. *See, e.g.*, CERT Coordination Ctr., Carnegie Mellon Software Eng'g Inst., Denial of Services Attacks (Oct. 2, 1997), http://www.cert.org/tech_tips/denial_of_service.html.

182. *See* Barnes, *supra* note 7, at 298–99; s*ee also* Pinkney, *supra* note 9, at 65–67.

183. *See* CERT Coordination Ctr., *supra* note 181, at 41 (presenting a calculation similar to the one in this paragraph).

exposed to externalities in security.[184] Others with valuable databases and sensitive files have strong incentives to protect their networks from attack in order to guard their investments and their reputations.[185] For example, IDC reports that "the recent surge in phishing attacks has created a sense of urgency within financial institutions, large internet service providers (ISPs), security technology providers, law enforcement agencies, and even university research labs."[186] The expenditures on security reported above suggest that many firms are in fact spending a considerable amount to protect their networks. Of course, not all firms will do so. Indeed, many small and medium companies may not have the personnel or resources to adequately protect their networks.[187] Any data that they expose could harm not only the attacked company's investment and reputation but also their customers whose files were stolen or misused.

Just as with individual consumers (A and B from above), we would expect companies to self-select into different security levels. The A businesses represent the uninformed, who do not patch because they do not know they need to. The B businesses represent the informed, who may sometimes choose to not patch when the costs of doing so outweigh the benefits. How many small to medium companies are like person A and ignore or are unaware of security issues, and how many are like B, taking what precautions they can, is unclear. The higher the stakes, however, the higher the probability that even small firms are aware of the risks and thus have incentives to take at least some precautions. Admittedly, externalities in software system security do exist: individuals and companies who do

184. This is the route that Randal Picker proposes for all critical infrastructure despite his recognition that air gapping, which he terms "autarky," would not solve the cyber crime problem. *See* Picker, *supra* note 5, at 26.

185. *See* Sam Y. Chung, Thomas Schneeweis & Kristina Eneroth, Corporate Reputation and Investment Performance: The UK and US Experience (Apr. 1999), *available at* http://papers.ssrn.com/sol3/papers.cfm?abstract_id=167629; *see also* KOLODGY, WORLDWIDE FIREWALL SOFTWARE, *supra* note 178.

186. BURKE & RYAN, *supra* note 86, at 4.

187. *See* ALLAN CAREY, IDC, INFORMATION SECURITY CONSULTING AND IMPLEMENTATION SERVICES 2005-2009 FORECAST 2 (2005) ("There still remains a shortage of [security] professionals who possess both information security and business acumen skills to effectively communicate with executive management."). *But see* KOLODGY, *supra* note 150, at 19 ("[V]endors that have historically been geared toward large enterprise are now developing product offerings geared to the small to medium-sized enterprises.").

not protect their networks expose others to some level of harm.[188] Nonetheless, companies and individuals who do value security can reduce the effects of externalities on their systems. Moreover, those persons and firms with valuable data have strong incentives to protect their networks, explaining the growing market in security prevention tools and the evident pressure on the larger developers to produce more security software.

Recent marketplace developments are making it easier for software developers to produce more secure software. A number of software tools for automated security checking during software development are available.[189] One such tool, Rough Auditing Tool for Security[190] (RATS), scans source code in a number of programming languages and flags common security related programming errors, including the buffer overflow problem described earlier. The tool offers suggestions for fixing some problems. RATS then rates the security vulnerabilities identified so that programmers can work on the highest priorities first.

Automatic patch services provide another marketplace innovation aimed at lowering the cost of maintaining a secure system for businesses and consumers. Many firms now offer automatic patch services that download and install relevant patches with a single acceptance click from the user. Such services lower the time cost of keeping security measures current.[191] For example, McAfee provides a subscription service for downloading software updates to neutralize the latest

188. Picker, *supra* note 5, at 34 ("The network is a sea of computing externalities, many extraordinarily positive but others that can range from everyday bothersome to enormously disruptive.").

189. Lana Gates, *A Sampling of Automated Testing Tools*, APPLICATION DEV. TRENDS, May 1, 2004, http://www.adtmag.com/article.asp?id=9308 (listing automated testing tools, such as AQtime, TestComplete, OneSight, and XDE Tester).

190. Secure Software, Download RATS, http://www.securesoftware.com/resources/download_rats.html.

191. Note that these services are market solutions, further casting doubt on the lack of patching stemming from an undervaluing of security. Companies such as Microsoft, McAfee, and Symantec offer automatic update programs to help ensure that businesses and consumers download the latest patches. For more on Microsoft, McAfee, and Symantec's automatic update programs, see McAfee, McAfee Upgrade Center, http://us.mcafee.com/root/upgradeCenter.asp (last visited Feb. 10, 2006); Microsoft, Microsoft Update: Help Keep Your Computer Current (Aug. 9, 2005), http://www.microsoft.com/athome/security/update/msupdate_keep_current.mspx; Symantec, http://www.symantec.com/index.htm (last visited Mar. 7, 2006).

spyware and viruses.[192] Similarly, Microsoft offers a free service for downloading patches to correct security weaknesses in Windows.[193] In its 2005 survey, CSI speculates that automated patching subscriptions are responsible for the declining number of respondents who say they install patches after a computer intrusion.[194] In 1999, nearly 100% of IT security officials surveyed responded that they patch in the wake of an intrusion.[195] In 2005, that figure was down to 73%.[196] When patching is automated, end users need not identify holes and install patches themselves, so a decrease in this statistic could reflect increased security.[197] A decrease in the total cost of patching should lead to an increase in patching activity, without requiring any change in end users' valuation of security. Nevertheless, the growing presence of security testing tools for developers and subscription services for users suggests end users—individuals and businesses—do increasingly value security.

Other private sector initiatives corroborate the observation that businesses value security. One such development is centered on catching corporate attackers and reporting attacks to the proper authorities. The Cyber Incident Detection and Data Analysis Center (CIDDAC) is a newly formed private company that provides "real-time cyber attack detection sensors."[198] CIDDAC installs software on corporate clients' websites that looks and reacts like a real entrance to a company's network—only it is not a real entrance. These false doors do not attempt to attract malicious hackers, but, rather, function to track any attackers who mistake the false doors for real doors. Given the prevalence of automated hacking tools, false doors can be a very effective means of identifying threats.[199] CIDDAC records

192. McAfee, *supra* note 191.

193. Aaron Ricadela, *Microsoft To Broaden Security-Patch Software*, INFO. WK., Mar. 16, 2004, http://www.informationweek.com/story/showArticle.jhtml?articleID=18400479.

194. CSI/FBI 2005 SURVEY, *supra* note 84, at 18.

195. *Id.* at 19.

196. *Id*.

197. One potential downside to automated patching is the risk that such services could be subverted to spread viruses instead of combat them.

198. Carole Moore, *Protecting Your Backdoor: The Cyber Incident Detection and Data Analysis Center Helps Law Enforcement Protect Its Private-sector Networks*, LAW ENFORCEMENT TECH., June 1, 2005 at 80, 87.

199. Law enforcement officials and non-profits employ a similar tactic, referred to as a "honeypot." A honeypot is a false website without an advertised name and

and analyzes an entry, determines where an attack originated, notifies the company as soon as an attack is detected, and passes the attack information on to law enforcement officials without identifying the company or industry that was attacked. In this way, the system protects the anonymity of its clients while still feeding attack information to cyber crime authorities. Third-party reporting of this type could prove instrumental in overcoming cyber attack victims' reluctance to report incidents for fear of bad publicity.

Thus, while market failures in the provision of software system security are apparent, not all failures are significant for all parties. Further, some market failures appear to be diminishing in significance due to emerging market solutions. The available data suggests that parties with truly valuable networks and files have strong incentives to take precautions to safeguard their systems. An asymmetric information problem for individual consumers emerges as the more troublesome market failure. But without better data on the real costs at stake and the preventive efforts that end users are taking, the ability to assess the real state of market failures in the provision of software system security is limited.

III. THE LAW OF SOFTWARE SYSTEM SECURITY

Most legal scholars would likely agree that the liability rules governing the distribution and use of software remain unclear, even after some thirty years of debate. Software does not fall neatly into existing laws regulating physical products. It is an "information good"—a product whose value lies in the information it contains, not in the tangible form it takes. Furthermore, it is frequently unclear who can be held liable when software security problems arise. Liability rules are important because they affect behavior,[200] and, thus can be used to mitigate the effects of market failures. The lack of clarity over software liability in existing legal doctrine has been a key factor motivating legislative proposals aimed at improving software

---

with a black screen. Any visitors to such a site are likely to be automated—such as worms. Thus, the Honeypot sites capture information on visitors as a means of identifying and tracking malicious hackers. *See* Project.Honeynet.org, The Honeynet Project: About the Project, http://project.honeynet.org/misc/project.html (last visited Oct. 2, 2006).

200. *See generally* Brown, *supra* note 12.

system security. In this Part, we briefly discuss the liability rules applicable to software and then evaluate recently enacted legislation.

A. Assigning Liability

The legal scholarship on assigning liability for software security problems is extensive and largely unsettled.[201] As economists, we do not intend to challenge that literature. We do, however, offer some observations rooted in the economics of software development. First, we observe that software creation, distribution, installation, and use involve a large number of disparate parties. As noted earlier, software system security depends on the actions of many, if not all, of these parties. Thus, if liability is disproportionately placed on any one of these parties, it could result in weaker overall security by undermining incentives to take precautions. Second, the nature of software has evolved from purely custom software embedded in hardware, to mass-distributed packaged software that could more easily be considered equivalent to traditional physical goods, to software services distributed over the internet. Software, accordingly, represents something of a moving target, making liability rules more difficult to settle.

Software system creation can help explain why there is ambiguity in the legal rules and court opinions governing software. Everyone from the software writers, to the person who installs that software, to the internet service provider that links the software to the Web, has a hand in creating a software system or network. Programming vulnerabilities can lead to security breaches, but so can improper program installation or

---

201. *See* Dierks, *supra* note 123, at 308 (pointing out that the debate is generally split into two camps: "rethinking old law, or creating new law"). The first camp argues that laws are generally flexible; it is just a matter of figuring out how the new technology fits into their framework. *See id.* The second camp argues that applying old laws to new technology is like forcing square pegs into round holes. *See id.*

In the software debate, Keith Hylton falls into the first camp: "[T]he theories reflected in tort doctrine are general and ought to apply without any serious modifications to cyberspace torts. There is no need for a special field of cybertort law." Keith Hylton, *Property Rules, Liability Rules, and Immunity: An Application to Cyberspace* 5 (Boston U. Sch. Of Law, Law & Econ. Working Paper No. 06-19, 2006). Pamela Samuelson, who supports the Semiconductor Copyright Protection Act, falls into the second camp. *See generally* Pamela Samuelson, *Creating a New Kind of Intellectual Property: Applying the Lessons of the Chip Law to Computer Programs*, 70 MINN. L. REV. 471 (1985).

maintenance by corporate IT staff, careless password protection by end users, and improper security procedures by internet access providers. Even if the software supplier provides a perfectly secure program, any one of the other involved parties can open a program or system to attack—knowingly or unknowingly. As a result, many security experts believe no system can ever be completely safe.[202] A hacker with enough skill can crack any system, using either code or social engineering.[203]

Given the complex set of interdependencies involved in software development, it is not always immediately apparent where to attach liability. A number of options might be available, such as contract rules, product liability, strict liability, no-fault liability, or negligence standards. Different liability rules apply to goods and services. Accordingly, the best approach for software systems depends, in part, on whether software is more like a traditional good, such as a clock radio, or more like a service, such as hiring a CPA.[204] Most scholars would argue that mass-produced consumer software, like the shrink-wrapped boxes of TurboTax sold each spring, are goods.[205] Other kinds of software, including software applications licensed over the internet and rented as needed, are more like services. In fact, the concept of software as a service is gaining momentum for large corporate software upgrades and could become a more important segment of the greater software market.[206]

Some legal scholars have argued that various types of software could be held to a strict liability standard.[207] Pamela

---

202. As one author writing on security issues expresses it, "[N]o systems are or will be 100 percent secure . . . ." Walter S. Baer, *Rewarding IT Security in the Marketplace* 4 (paper presented at the Research Conf. on Comm., Info, and Internet Pol'y, Sept. 21 2003), *available at* http://tprc.org/papers/2003/190/BaerITSecurity.pdf (*earlier version printed in* 24 CONTEMP. SECURITY POL'Y 190 (2003)); *see also* Michael Lee et al., Comment, *Electronic Commerce, Hackers, and the Search for Legitimacy: A Regulatory Proposal*, 14 BERKELEY TECH. L.J. 839, 854 (1999).

203. *See* Lee et al., *supra* note 202, at 855–64.

204. *See* Warren E. Agin & Scott N. Kumis, *A Framework for Understanding Electronic Information Transactions*, 15 ALB. L.J. SCI. & TECH. 277, 298–99 (2005).

205. *See, e.g.*, Skibell, *supra* note 75, at 122–23.

206. In 2003, the top three vendors of software as a service collected revenues of $366.9 million, representing nearly double-digit or double-digit revenue growth from the previous year for each vendor. AMY MIZORAS KONARY, IDC, WORLDWIDE SOFTWARE AS A SERVICE 2003 VENDOR SHARES: SAAS AND ENTERPRISE ASP COMPETITIVE ANALYSIS 6 tbl.2 (2004).

207. *See* Hylton, *supra* note 201, at 38–39 ("Strict liability differs from negligence in the sense that it affects activity levels. It forces the liable party to think about

Samuelson, for example, argues that strict liability should apply to software embedded in components of equipment—such as airplanes, X-ray machines, and automobiles.[208] This point has yet to be tested in court.[209] Regardless, a considerable amount of software is not embedded in equipment, leaving the liability question open for most software uses.

Other legal scholars have noted that software could arguably be treated more like hardcopy books, which are not defined as "goods" for the purpose of determining liability.[210] Although books have a physical manifestation, their value resides in their information content.[211] Wary of unduly restricting the free flow of information, courts have limited information liability to defamation or erroneous statements that defraud end users of books.[212] Software, especially custom software and software as a service, could also be viewed as having its value reside in its information content. Thus, the courts may decide to treat software as they do books by employing a less stringent liability rule.

If the courts consider software an information good, software production might be held to a negligence standard. Professional information providers, including doctors and lawyers, can be held liable for negligence.[213] This standard requires one to offer himself as an expert. Nevertheless, even for this case of well-established law, proving malpractice is difficult because of differences of opinion among qualified professionals.[214] Fur-

---

how frequently it wishes to engage in the activity giving rise to liability, or to think about deep design changes that would reduce the frequency of injuries even when the activity is undertaken with optimal care.").

208. Pamela Samuelson, *Liability for Defective Electronic Information*, COMM. OF THE ACM, Jan. 1993, at 21.

209. There is at least one precedent-setting case that could be used to apply strict liability to mass-produced consumer software. In *Aetna Casualty & Surety Co. v. Jeppeson & Co.*, 642 F.2d 339, 341–42 (9th Cir. 1981), the court found that an aeronautical chart was mass-produced for commercial purposes and that people using the chart were relying on the mapmaker's expertise. *See* Samuelson, *supra* note 208, at 23–24.

210. *See* Samuelson, *supra* note 208, at 22–23; *see also* Winter v. G.P. Putnam's Sons, 938 F.2d 1033 (9th Cir. 1991); Cardozo v. True, 342 So. 2d 1053, 1057 (Fla. Dist. Ct. App. 1977).

211. Samuelson, *supra* note 208, at 22.

212. *See id.*, at 23.

213. *See* i*d.*, at 24–25.

214. *See* Gregory E. Maggs, *Consumer Bankruptcy Fraud and the "Reliance on Advice of Counsel" Argument*, 69 AM. BANKR. L.J. 1, 28 (1995) (addressing the difficulty in proving malpractice suits).

thermore, software developers are neither generally considered to rise to the level of expert "professional," nor are they licensed like doctors, lawyers, and accountants.[215] Thus, a negligence standard could be more difficult to apply to software development.

The ambiguity over applicable legal rules for software development has greater implications for security policy. Most importantly, without defined legal rules or standards, victims of software security breaches have not known where to turn for redress. Suits against software developers, internet access providers, and insurance companies have all been brought in court with varying degrees of success.[216] Without clear lines of liability, policymakers often feel pressured to craft new laws to make up for perceived holes in the existing ones. Thus, the very nature of the product and its commonality with information goods have been primary factors pushing for changes in software system security policy.

B. Recent Software System Security Legislation

A review of existing laws suggests that policies aimed at improving software security have not been terribly effective. Table 4 summarizes the key federal laws aimed at improving information technology security over the last twenty years and an influential state law that may inspire a federal version. Many of these laws, especially the earlier ones, extend traditional physical world policy to virtual world crimes. In other words, many of the bills focus on prosecuting criminals after a cyber crime occurs rather than on preventing cyber crimes from occurring.

---

215. Samuelson, *supra* note 208, at 24–25.

216. *See e.g.*, Green v. America Online, 318 F.3d 465 (3d Cir. 2003); Lucker Mfg. v. Home Ins. Co., 23 F.3d 808 (3d Cir. 1994); Centennial Ins. Co. v. Applied Health Care Sys., Inc., 710 F.2d 1288 (7th Cir. 1983); Hamilton v. Microsoft Corp., No. BC303321 (Cal. Super. Ct. Apr. 13, 2004).

Table 4: Key Federal and State Legislation

Act	Description
Computer Fraud and Abuse Act of 1986 (CFAA), Pub. L. No. 99-474, 100 Stat. 1213 (1986) (codified as amended at 18 U.S.C. § 1030).[217]	Criminalizes unauthorized computer use; explicitly avoids design and production defect liability.
Uniting and Strengthening America Act by Providing Appropriate Tools Required to Intercept and Obstruct Terrorism Act of 2001 (*USA PATRIOT Act*), Pub. L. No. 107-56, 115 Stat. 272 (2001).	Grants federal agencies "authority to intercept wire, oral, and electronic communications relating to computer fraud and abuse offenses." Includes section titled "Deterrence and Prevention of Cyberterrorism."
Cyber Security Research and Development Act of 2002 (CSRDA, Pub. L. 107-305, 116 Stat. 2367 (2002).	Authorized $903 million over five years for new research and training by the National Science Foundation and the National Institute of Standards and Technology with the aim of developing ways to prevent and respond to terrorist attacks on computers.
Federal Information Security Management Act of 2002 (FISMA), Pub. L. 107-347, 116 Stat. 2899 (2002).	Requires all government agencies to secure information and information systems that support their operations.
The Identity Theft Penalty Enhancement Act of 2004 (ITRPEA), Pub. L. 108-275, 118 Stat. 831 (2004).	Defines identity theft, sets mandatory prison sentences for identity theft, and provides increased penalties for identity theft associated with terrorism.
California Security Breach Information Act (S.B. 1386), 2002 Cal. Stat. 915.	Mandates public disclosure of computer-security breaches if confidential information of a California resident may be compromised.

Most of the above-listed laws are aimed at defining online criminal behavior and setting sentence terms for those caught.[218] Although punishments can act as a deterrent, these

217. Pinkney notes that this effort was Congress's reaction to the 1983 tech-cult-classic movie *War Games*, in which a high school boy, while looking for advanced copies of video games, stumbles across the system the Department of Defense uses to control nuclear missile silos. *See* Pinkney, *supra* note 9, at 62.

218. Dierks argues that:

> [F]aith in the ability of computer crime laws to address adequately the problem of computer abuse is misplaced . . . [in part because] at least six factors make the ex post criminalization of computer network abuse

laws have had little real impact thus far.[219] For instance, in the first six years of CFAA history, only one successful prosecution was brought—Robert Morris was convicted for writing the first computer worm.[220] Prosecutions increased through the 1990s, but were still relatively rare. Only 83 cases were prosecuted in 1998.[221] Of those prosecuted, 47 were convicted and only 20 of those convicted served any prison time, with an average sentence of 5 months.[222] The DOJ declines to prosecute 64% to 78% of the referred cases, largely because of a lack of evidence.[223] The internet lends itself to anonymity and geographic agility,[224]

> problematic: (A) the presence of arbitrary spatial distinctions in cyberspace; (B) the difficulty of detecting criminal activity in cyberspace; (C) the difficulty of determining criminal identity in cyberplace; (D) the difficulty of proving criminal culpability in cyberplace; (E) the absence of incentives to report cyber crime; and (F) the absence of deterrence in present criminal law provisions.

Dierks, *supra* note 123, at 330–31.

219. Many of the laws listed in Table 4 are controversial for other reasons as well. A number of them were passed quickly in the wake of September 11, 2001. For example, the USA PATRIOT Act has attracted criticism from civil liberties groups—both at the time of passing and more recently during the renewal debate—over certain provisions granting the government "sweeping" powers, especially wiretaps. *See, e.g.*, Ellen S. Podgor, *Computer Crimes and the USA PATRIOT Act*, CRIM. JUST. MAG., Summer 2002, at 61; Matthew Purdy, *Bush's New Rules to Fight Terror Transform the Legal Landscape*, N.Y. TIMES., Nov. 25, 2001, at B4. By 2006, acrimonious debate was in full swing. In mid-November 2005, the negotiations over reauthorization of the bill broke down, because of concern that the bill did not strike the proper "balance between fighting terrorism and protecting civil liberties." Eric Lichtblau, *Congress Nears Deal to Renew Antiterror Law: Sweeping U.S. Powers Are Mostly Preserved*, N.Y. TIMES, Nov. 17, 2005, at A1. By early 2006, there were charges that President Bush's acts related to wiretaps might have been illegal. *See* Eric Lichtblau & Adam Liptak, *Bush and His Senior Aides Press On in Legal Defense for Wiretapping Program*, N.Y. TIMES, Jan. 28, 2006, at A12.

220. United States v. Morris, 928 F.2d 504 (2d. Cir. 1991).

221. David Banisar, *Computer Hacker's Sentence Spotlights High-Tech Crime Prosecutions*, CRIM. JUST. WKLY. Aug. 3, 1999, at 225.

222. *Id.* The article notes that from 1992 through 1999, 196 persons were convicted and 84 persons were sentenced to prison under the Computer Fraud and Abuse Act. More recent data is difficult to obtain, requiring individual case reviews obtained through the Freedom of Information Act. Sentencing may be toughening up, however. In December 2005, one conviction carried a nine year prison term. Brian Grow, *Hacker Hunters*, BUS. WK., May 30, 2005, at 77.

223. *See* Banisar, *supra* note 221.

224. *See* Grow, *supra* note 222. Note, however, that bug bounty programs, discussed below, provide monetary incentives for insiders to turn hackers over to the authorities. *See* Robert Lemos, *Microsoft's Bounty Hunter*, CNETNEWS.COM, June 10, 2004, http://news.com.com/Microsoft%27s+bounty+hunter/2008-7355_3-5228216.html (reporting that a $250,000 reward from Microsoft enticed an informant to disclose information about the person who wrote and released the Sasser worm).

making cyber criminals far more difficult to catch than criminals in the physical world. Moreover, cyber criminals frequently operate outside of U.S. jurisdiction.[225] A computer attack can involve multiple computers, many of them zombies, making it difficult to trace a malicious hacker. Computer criminals can also easily use an anonymous account at a cyber café or use software to cover their tracks.[226] As a result, many malicious hackers are never caught. Douglas Barnes explains a consequence of online anonymity: "There is, then, a justifiable perception among worm authors that only exceptionally careless authors get caught, and this causes authors to deeply discount the occasional law enforcement success."[227]

A few of the legislative acts listed in Table 4 have the potential to make meaningful contributions to software system security. The Cyber Security Research and Development Act could lead to the acquisition of more reliable data on software system security, although its focus is narrow. FISMA also has potential. If implemented appropriately, it could lead to better security within government agencies and provide those agencies with valuable experience in improving software system security. This experience could be shared with the private sector. So far, critics of FISMA argue that the annual security audits it mandates have done little more than fund consultants and generate reports, "shelfware," that are subsequently ignored.[228] The average agency grade was a D+ in 2004, two years after the Act passed.[229] Nonetheless, some consider the process of securing networks evolutionary, arguing that the Act has led to a

---

225. *See generally* Grow, *supra* note 222.

226. For example, "Hackers sitting in a wireless 'hot spot' can gain access to an employee's laptop or can hack a wireless provider's servers, such as in the case of T-Mobile, where a black hat [malicious hacker] was able to download sensitive government documents." McComas, *supra* note 62, at 84.

227. Barnes, *supra* note 7, at 283. Barnes notes that "expert Sarah Gordon surveyed virus authors and IT security professionals following 'Melissa' virus author David Smith's arrest. The eleven virus authors unanimously agreed that the arrest would have no effect, whereas the sixteen security professionals were evenly split." *Id*. at 283 n.20 (citing Sarah Gordon, IBM Thomas J. Watson Res. Ctr., Virus Writers: The End of Innocence?, http://www.research.ibm.com/antivirus/SciPapers/VB2000SG.htm). Barnes provides examples of the careless mistakes that have led authorities to worm authors: "For instance, the author of a variant of the 'Blaster' worm was recently caught because Romanian language text in the worm led police to a webpage that listed his address and phone number." *Id.* at 283.

228. *See* Benton Ives-Haperin, *Effectiveness of Cybersecurity Law Questioned*, CQ HOMELAND SECURITY, Nov. 15, 2005.

229. *Id*.

better risk-based approach to improving security that will pay off eventually.[230]

Finally, California's mandatory reporting law has been praised as crucial in raising public awareness of security issues.[231] As noted before, companies are frequently reluctant to disclose security breaches for fear of negative publicity. The California law provides them no choice when personal records are violated. In the wake of the ChoicePoint incident, several other states introduced bills modeled after California's law.[232] In fact, California state senator Deborah Bowen argued that consumers across the nation would not have known about the ChoicePoint breach had it not been for California's law.[233] Increased reporting, as noted earlier, could be a key to reducing information asymmetry. When software developers, internet service providers, and other companies charged with safeguarding sensitive data are subjected to bad publicity after each breach, there are increased reputation-based incentives to take safety precautions before a breach occurs.

As the above discussion makes clear, only two laws in Table 4 address either of the market failures put forth for software system security. The first is California's mandatory reporting law. That law aims to provide the public with improved information on security breaches, although the law does not directly address the information asymmetry in software or network security provision. The second is FISMA, which attempts to solve the security externality problem among government agencies. The jury is still out on whether FISMA will eventually be successful. The remainder of the acts are largely hasty reactions to September 11th, aimed at making cyber terrorism illegal and increasing the flexibility law enforcement agencies have in pursuing terrorist related cases. Given the dearth of existing policies for effectively improving software system security, the interest in new proposals is understandable. The following Part reviews many of the more prominent of those proposals.

---

230. *See* Matthew Weigelt, *Davis: FISMA Could Prevent "Cyber Pearl Harbor,"* FCW.COM, Apr. 27, 2006, at http://www.fcw.com/article94211-04-27-06-web.

231. *See generally* Laura Mahoney et. al., *ChoicePoint Incident Prompts State Lawmakers to Offer Data Notification Bills*, 10 ELECTRONIC COM. & L. REP. 217 (2005).

232. *Id.* at 217–18 (noting the introduction of bills in Florida, Georgia, Illinois, Minnesota, New York, Rhode Island, Texas, and Washington).

233. *Id.*; *cf.* McComas, *supra* note 62, at 85 (arguing that if the threshold for notification is set too low, "alarm bells of security breaches would be ringing so often the public would become desensitized to the threat").

IV. THE FUTURE OF SOFTWARE SYSTEM SECURITY

The various types of software system security problems discussed at the beginning of this paper suggest the need for a multi-faceted response. For instance, denial of service attacks are aimed at internet portals and servers linked to public networks. It can be argued that internet access providers and companies with servers on the internet should bear significant responsibility for preventing or limiting the damage that denial of service attacks can inflict.[234] Backdoors and unintentional design vulnerabilities, on the other hand, involve programmers and processes inside a software development company. Thus, the presence of backdoors and unintentional design vulnerabilities suggests an entirely different approach to improving security focused on software makers. Social engineering data breaches, such as the one at ChoicePoint, call for yet another approach, one aimed at security training for companies with sensitive datasets, especially for employees with access to customer data. Now, we review some of the more prominent policy suggestions from the legal literature along with a few ideas of our own.

Table 5 provides a summary of proposals. In evaluating possible initiatives, we employ the economic framework presented earlier. Viewed from the standpoint of economics, or just plain common sense, many of the proposals to date are hard to justify; they either fail in terms of a basic cost-benefit framework, or they have serious problems in design. For example, several proposals do not adequately consider the adverse, unintended consequences that could result from a change in the law.

---

234. *See, e.g.*, Lichtman & Posner, *supra* note 156, at 3 (arguing that "rules that hold one party liable for the wrongs committed by another are the standard legal response in situations where, [as with internet service provision], liability will be predictably ineffective if directly applied to a class of bad actors . . . ."). On the other side of this debate, Professor Neal Katyal maintains that if "the burden for crime prevention is placed on ISPs, so that they are responsible for the criminal acts of their subscribers, the result will be harm to the Net and its users as ISPs purge their subscriber base of customers who arouse the faintest whiff of suspicion." *See* Katyal, *supra* note 4, at 2282.

Table 5: Summary of Proposals for Improving Software System Security

Category	Proposal	Evaluation
Regulating software developers	• Enacting a "lemons law" for software	• Would be difficult to make practical
	• Holding software makers liable for all damages caused by exploited security holes	• Strict liability not justified as it could *reduce* overall security level by eliminating incentives for other parties to take precautions, but some liability with damage thresholds might be beneficial
	• Mandating performance standards for software	• Technology moves quickly so the government could not keep up; possible negative effect on innovation
	• Prohibiting the outsourcing of software for certain government agencies	• Outsourcing poses risks, but it is unclear that regulation is needed
Regulating software users	• Mandating use of security features in software	• Would be difficult to enforce, but could be used to mitigate damages in lawsuits
	• Mandating security breach reporting	• Could help increase information available, especially if reporting is anonymous
Regulating cyber weapons	• Tagging software	• Unlikely to reduce attacks
	• Banning viruses and other malware from the internet	• Unlikely to reduce attacks
Government leading by example	• Purchasing only secure software	• Might provide further incentives for secure software design

	• Testing practical implementation of legislation on own agencies	• Would enhance the government's credibility on security issues and provide information on likely impact of legislation
	• Work with other nations to reform the Common Criteria software security review process	• Could improve the international testing organization
Software Developer Initiatives	• Bug bounties	• Moderately successful and cost effective
	• Revising software development process to incorporate security	• Appears to be effective; some larger companies are implementing
Cyber insurance	• Encouraging or mandating cyber insurance	• Shows great promise in reinforcing the right incentives, but the market is developing very slowly

Sources: See text.

A. Regulating Software Developers

Lemons Law: Academic literature suggests a number of new legislative efforts aimed at software developers. For example, Barnes is supportive of government intervention for "deworming" the internet; he finds that, "to the extent that this concern appears on the agenda of policymakers, they apparently assume that markets will eventually provide the right incentive for software publishers to produce better software. This has not happened."[235]

Barnes suggests that a lemons law for software is one solution to perceived market failures in preventing worms on the internet.[236] If a software developer, "who could have prevented worms by choosing methodologies and technologies that can achieve a high level of worm resistance" chooses not to use those methodologies and technologies, then the lemons law would require his firm to provide refunds and information dis-

235. Barnes, *supra* note 7, at 281.
236. *Id.* at 282.

closures to purchasers.[237] We can consider this type of law in the context of the simple model presented earlier. Creating less secure software would run the risk of exposing developers to significant expenses in the form of refunds and the subsequent redevelopment of software. Developers would compare these expected costs against the increased cost of developing more secure software. Security would improve so long as the expected costs associated with distributing lower-security software exceeded the increased cost of developing higher-security software.

Proponents of a lemons law often point to before and after cost estimates to emphasize the need for more "before" incentives for software developers. For example, *Secure Business Quarterly* estimates that it costs a software developer around $24,000 to catch and fix a bug during the initial testing phases for a new program, compared to $160,000 to catch and fix that same bug after the software has been deployed.[238] Ex post statistics, however, can mask important aspects of the software development process. Hylton argues that one cannot compare the cost of fixing a security flaw after the fact with the cost of fixing that same flaw during the development phase: "Presumably any competent programmer could correct a flaw, viewed in isolation, once it has been identified."[239] The trick lies in identifying the flaws. Typical software testing can only go so far. "[T]esting can never prove the absence of fatal flaws in software. Testing can at best establish that the program is not likely to fail under certain uses."[240]

The limits of traditional testing—tacked on to the end of the software creation process—point to the importance of incorporating security into the full life cycle of software development. But moves in this direction, with "a culture change away from the cowboy and toward the engineer,"[241] are already underway due to market forces.

---

237. *See id*.

238. Eric Karofsky, *Defining the Value of Strategic Security*, 1(2) SECURE BUS. Q. (2001).

239. Hylton, *supra* note 201, at 44.

240. Gemignani, *supra* note 158, at 191.

241. Lohr, *supra* note 168 (quoting Shawn Hernan, security specialist at CERT); *see also* JODY ARMOUR & WATTS S. HUMPHREY, SOFTWARE PRODUCT LIABILITY 13 (Carnegie Mellon Univ. Software Eng'g Inst. No. CMU/SEI-93-TR-13, 1993) ("[N]o self-respecting semiconductor engineer would consider testing and fixing all the defective chips coming off the production line.").

In addition to market developments, it is important to note that the lemons law analogy does not translate easily to software. Traditional lemons laws apply to faulty manufacture, not faulty design. As discussed earlier, apart from the odd dysfunctional compact disc, each copy of a software program is identical. Therefore, unlike in the used car setting, if a lemons law were applied to software it would be design-based and all copies of a program would have to be recalled. After providing refunds to every purchaser, the developer would need to either start over with a new product or exit the business. On the other hand, most lemons laws stipulate exceptions that could also apply to the typical software situation. Although most consumers do not alter the actual code, installing a program on a computer where it will interact with other resident programs and be exposed to online downloads might make it difficult to isolate the origin of a security flaw. More importantly, pushing full liability onto developers would reduce users' (and ISPs') incentives to take adequate precautions to protect their systems.

Liability Rules for Software Developers: Pinkney suggests a modified liability rule that recognizes the difficulty in providing completely secure software.[242] Under his proposal, software makers would be strictly liable for all damages caused by security holes exploited in their products.[243] Providing patches for the security holes, however, would provide a defense for the software developers. In this regime, security engineering by software developers would be complemented by their issuing ex post fixes for any problems not caught in the creation phase. End users would be responsible for installing patches. Developers would be liable only for the period of damage from the first attack to the first patch available. Since developers would be liable for the actual damages incurred, this proposal could match security efforts to economic harm and would thus avoid the overinvestment in security that would likely result from a broad lemons law.[244] The limited liability approach, however,

---

242. *See* Pinkney, *supra* note 9, at 79.

243. *See id*.

244. As Randal Picker argues, "[f]ull-blown liability would help solve a software adoption version of the prisoner's dilemma—each user wants the other user to adopt early and get bugs out of the system—but would also introduce standard adverse selection problems." Picker, *supra* note 5, at 9. Full liability would lead to over-consumption of software by end-users with high costs from malicious hack-

would involve litigation, which could be costly. Moreover, this approach might encourage software developers to rush the release of a patch. If not fully tested, a patch might introduce more security vulnerabilities than it corrects. Further, care is necessary to reduce or eliminate damage awards when users or other responsible parties take actions that compromise the security of the software system. The final rule would need to carefully balance responsibility across all parties involved.

Finally, any liability rule for software should help to reduce the overall social costs. Recall from the economic framework that an efficient policy should minimize the sum of damages plus prevention costs. As noted above, ignoring minor worms may be a rational strategy for both software developers and consumers. If the damages imposed by a particular worm are small, then it is likely that prevention costs would outweigh the benefits of reducing the number of worms on the internet. Therefore, it may be desirable to modify any liability proposal to require a threshold of damage before demanding restitution. For serious security breaches, such as vulnerabilities that enable data theft or corruption, the benefits of requiring developers to pay fines may justify the costs, but it is difficult to know without better data. Consequently, we believe that strict liability is not justified at this time. A law that contains a damage threshold, in which fines are reduced or eliminated when users do not take adequate precautions, could be worth considering.

Performance Standards: A different approach would set performance standards. Analogous to building code laws intended to correct for safety information asymmetries and public health issues, Katyal suggests mandating performance standards for software as a means of correcting security information asymmetries and network security issues.[245] The standards would not specify the exact technologies to be used so as to remain flexible.

The problem with such proposals is that they do not consider the difficulties of defining standards in a rapidly evolving industry. Even without the government codifying required technologies, maintaining appropriate standards would be difficult given the pace of software development. The government could name an industry association or an independent over-

---

ing. Consumers would also have less incentive to exercise caution in installing and using software under a full liability regime. *See id.* at 28.

245. Katyal, *supra* note 4, at 2286.

sight group to be responsible for the actual standards, but some sort of enforcement mechanism would be needed to ensure compliance. Timing issues would pose problems as well. Software written a year ago would need to be judged under last year's standard to avoid "perfect hindsight" problems, but upgrades could pose a unique problem. Would the upgrade alone be held to the higher standard of today? For software with multiple versions, one could imagine a long series of performance standards that could apply to a single program, meaning all changes would need to be carefully tracked and dated. If each upgrade moved the entire program to a newer standard, a law of this sort would likely affect innovation and research investments, resulting in delayed releases and fewer upgrades. In the end, such a proposal would be both hard to implement and likely to impose serious unintended costs.

Business Practice Restrictions: Some proposals for regulating software development firms have focused on the common practice of outsourcing. Software developers frequently contract for development work done outside of the United States. Some of those contracts are in countries with known terrorist networks and could, therefore, represent security threats in the form of backdoors or Trojan horses inserted into programs.[246] Banning outsourcing altogether would be an extreme measure, costing firms significant amounts in increased labor costs.[247]

Less draconian measures than an outright ban are worth consideration. For example, companies developing custom software for certain government agencies—such as the De-

---

246. As Clay Wilson of the Congressional Research Service observes:

> Oracle, a major database software vendor and a supplier to U.S. intelligence agencies, has in the past contracted for software development in India and China. Terrorist networks are known to exist in other countries located in Southeast Asia where some contract work has been outsourced, such as Malaysia and Indonesia. Other possible recipients of outsourced projects are countries such as Israel, India, Pakistan, Russia and China.

CLAY WILSON, COMPUTER ATTACK AND CYBER TERRORISM: VULNERABILITIES AND POLICY ISSUES FOR CONGRESS, RL32114, at 22 (2003).

247. According to a McKinsey Quarterly report, U.S. companies save 58 cents for every dollar spent on business services that move offshore. Justin Marks, *The Outcry Over Outsourcing*, 30 ST. LEGISLATURES 30 (May 2004), *available at* 30, *available at* http://www.ncsl.org/programs/pubs/slmag/2004/04SLMay_Outsourcing.pdf. The head of offshore operations in India for Electronic Data Systems notes that U.S. companies attain a cost savings of over 60% through Indian software services. *A New Battle Over Offshore Outsourcing*, BUS. WK. ONLINE, June 6, 2003, http://www.businessweek.com/smallbiz/content/jun2003/sb2003066_8575.htm.

partment of Defense and the Department of Homeland Security—could establish no-outsource rules, with strictly enforced ethical walls separating development teams for government projects from other teams within the software company. It is unclear, however, why a law is required to establish this kind of rule, because guidelines in requests for project proposals issued by government agencies would suffice. Individual agencies can also specify that off the shelf software can not be used for certain sensitive applications in order to maintain a higher security standard.

B. Regulating Software Users

Even if software developers improve security dramatically through their new emphasis on security-focused processes, software system security would still be vulnerable to end user practices. Therefore, some regulatory proposals have focused on individuals and firms consuming software.

Forbid Disabling of Security: Just as car owners are not allowed to disable the emission control devices on their automobiles, legislation could prohibit disabling or failing to use security features in software.[248] It would be impractical, however, to have each end user bring their computers into a testing facility (as end users of cars are often required to do). Nonetheless, a rule of this sort may be useful in conjunction with other liability rules. For example, users who do not implement developer-recommended security procedures or who do not install relevant patches in a timely fashion could be barred from obtaining damages for an otherwise covered security problem. But even here, determining whether end users followed proper procedures could be difficult and would require that specific actions be delineated in the law.

Mandated Reporting: Other proposals center on the lack of information on security breaches. As discussed above, companies that are victimized rarely report attacks to authorities such as the FBI. Better reporting by victims could help to locate the most serious vulnerabilities and further foster a reputation-based market for secure software providers. Given the benefits to be gained from increased reporting, some have called for mandatory incident reporting along the lines of the California law. Any federal laws requiring security incident reporting

248. Barnes, *supra* note 7, at 329.

would likely face stiff opposition unless the process could be made confidential.[249] One option would be a federal program that did not release information about individual victims but reported only aggregated data.

Although there are clearly benefits from mandatory reporting, there are also costs. Some studies indicate an "upswing in intrusions using a given security weakness once it has been publicly disclosed."[250] That is, reporting informs not only consumers and businesses, but also malicious hackers.

The emergence of private sector alternatives also weighs against mandatory federal reporting. As discussed previously, private firms exist that filter information so that anything identifying the victim company is removed before the information is given to the authorities. Increased reliance on third parties could provide solid data on security incidents while maintaining corporate anonymity at the same time. There might still be, however, a possible link between reporting an incident to authorities and encouraging others to engage in copycat attacks. Because determined malicious hackers will make a point of learning about vulnerabilities, the benefits to be gained through increased reporting would likely outweigh the costs.

C. Regulating Cyber Weapons

A third potential approach would be to focus on the tools employed in software system security breaches. Dorothy Denning explores the possibility of regulating the weapons themselves, without endorsing or condemning the approach.[251] The idea is an interesting one, but does not appear to offer enough benefits to offset the costs of enforcement.

Legislative precedent does not present an obstacle. In terms of online security, the law already makes writing a worm or virus illegal, similar to how it restricts the physical production of weapons.[252] Additionally, laws banning cell phone scanners, equipment for producing counterfeit currency, and software that circumvents copyright protection provide precedents for laws controlling software weapons that cause only economic

---

249. Baer predicts that mandatory reporting would be a "non-starter" politically. *See* Baer, *supra* note 202, at 17.

250. Skibell, *supra* note 75, at 115 (citing William A. Arbaugh et al., *Windows of Vulnerability: A Case-Study Analysis*, IEEE COMPUTER, Dec. 2000, at 52).

251. Denning, *supra* note 77, at 43–53.

252. Computer Fraud and Abuse Act, 18 U.S.C. § 1030 (2006).

harm, not physical harm.[253] Regulating cyber weapons, therefore, appears feasible from a legal standpoint. The question is whether it is advisable. Here we review a few of the options, again evaluating them from a practical implementation standpoint along with considering their costs and benefits.

Software Tags: One possibility for cyber weapon control is software tagging. Similar to the serial numbers on guns, a software tag could be embedded as a digital watermark.[254] Possession or distribution of software without a tag would be a criminal offense. Weighing against such an approach are both practical and ethical considerations. First, unlike physical guns, malware production is not limited to a handful of legitimate companies. The goal is not to keep legal weapons out of the hands of criminals. Instead, criminals are the ones creating the weapons. Given that they are already creating destructive tools to be used in defiance of one set of laws, there is little reason to expect that they would respect another set of laws and embed a tag into their malware.[255] On the ethical side, forced tagging could be viewed as an invasion of privacy, a means for governments to track online behavior.

Malware Ban: Another way of controlling cyber weapons would be to ban viruses and other harmful malware from the internet.[256] With an outright ban, companies with servers on the internet could automatically scan for such weapons, checking documents as they come and go. This tactic would encourage other parties, such as internet portal hosts, to spend additional resources on prevention efforts by assigning them some liability for keeping their servers free of banned programs. The government would only intervene when someone reported a banned program that a server's owner did not remove. A rule along these lines might help to reduce the number of active script kiddies by limiting the tools readily available to them, although it would likely do little to prevent skilled hackers from creating their own malware.

Perhaps the biggest hurdle in regulating cyber weapons is political. Truly effective control of cyber weapons would re-

253. *See* Denning, *supra* note 77, at 6.

254. *See id.* at 10.

255. Tagging might be useful, however, for software that can be used either for productive or destructive purposes. Tagging might also deter script kiddies, who rely on downloadable tools.

256. *See* Denning, *supra* note 77, at 12.

quire coordinated international efforts, perhaps requiring treaties and agreements on how criminal tracking and prosecution would work in practice.[257] Any treaties would need to be carefully crafted.[258] If enforced too zealously, cyber weapon control rules could reduce legitimate, welfare-enhancing software research and development.[259] Finally, Denning notes one last downside to this approach: "a cyberarms control treaty would have the disadvantage of limiting the ability of the U.S. and other nations to deal with adversaries."[260] Of course, international treaties and multi-country negotiations are a political reality in many arenas today, most notably in the fights against terrorism and drugs. Thus, if controlling software weaponry were made a priority, political obstacles could likely be overcome.

D. Government Leading by Example

Instead of legislating the activities of others, the government might have a greater impact on software security if it were to lead by example. One possible approach is for the government to encourage the development of more secure software by purchasing those products that are more secure. The government is a key buyer of software, consuming approximately forty-two percent of all software and computing services as measured by some statistics.[261] Certain highly sensitive departments already have internal rules in place. For example, the Department of Defense requires that all new software be submitted to the Na-

---

257. National governments have, however, succeeded in raising boundaries in several instances. *See* JACK GOLDSMITH & TIM WU, WHO CONTROLS THE INTERNET? 6–7 (2006).

258. *Id*. at 165–67.

259. Some cyber weapons, such as viruses and worms, are used mainly for destructive purposes; but other programs, such as password catchers and decryption programs, are capable of both benign and malicious uses.

260. *See* Denning, *supra* note 77, at 11.

261. According to IDC, the total IT security market for software amounted to $10 billion in 2004. BURKE ET. AL., *supra* note 26, at 1. Moreover, "Federal agencies spent $4.2 billion securing the government's total information technology investment of approximately $59 billion or about seven percent of the total information technology portfolio." OMB, FEDERAL INFORMATION SECURITY MANAGEMENT ACT (FISMA) 2004 REPORT TO CONGRESS (2005). The government's IT investment in 2004 was approximately 2.7% of the approximately $2.2 trillion 2004 proposed budget. *See* Press Release, White House, Fact Sheet: President Bush's 2004 Budget (Feb. 3, 2003), http://www.whitehouse.gov/news/releases/2003/02/20030203-6.html.

tional Security Agency for security testing.[262] In general, U.S. government agencies that handle "sensitive but nonclassified information" can use off the shelf software, but only if the software has passed Common Criteria's Evaluation Assurance Level 4, the security review process mentioned above.[263]

Most government agencies, however, pay little or no attention to security issues. The first annual review completed under FISMA was released in late 2003. The report examined the computer security practices of federal agencies and awarded those agencies grades for their overall IT security.[264] Over half of the agencies received a D or an F. Especially discouraging, the Department of Homeland Security, which has a division devoted to monitoring cyber security, received an F, as did the Justice Department, the agency responsible for investigating and prosecuting cyber crime. Taken as a whole, the twenty-four agencies surveyed averaged a D. A year later, that average was still only a D+. The Orange Book program discussed previously illustrates the underlying problem: security costs money and reduces features.

State governments would probably fare even worse if similar evaluations were conducted at the state level. As Larry Kettlewell, chief information security officer for Kansas, observes, "[t]here are not that many states that have, on staff, people with the level of expertise needed to keep up with the latest systems, and, along with that, who know how to make those systems secure."[265] The state-federal comparison therefore seems analogous to the consumer-business distinction drawn earlier.

The overall poor performance of the government on computer security issues makes proper implementation under FISMA a key issue. Mandating secure systems is not enough; agencies need a clear way to achieve that goal. Some of the agencies pressured to upgrade their security under FISMA have complained about ambiguity in the guidelines.[266] It is also

---

262. WEIMANN, *supra* note 67.

263. Alex Salkever, *Microsoft Earns a Security Merit Badge*, BUS. WK. ONLINE Nov. 5, 2002, http://www.businessweek.com/technology/content/nov2002/ tc2002115_4021.htm.

264. *See* WEIMANN, *supra* note 67, at 3 (citing a *Washington Post* study).

265. Boulard & Goodwin, *supra* note 134, at 25 (quoting Larry Kettlewell).

266. *See* Aliya Sternstein, *Interior Secretary Downplays FISMA Flaws*, FCW.COM, Nov. 21, 2005, http://www.fcw.com/article91521-11-21-05-Web (noting complaints by the Department of the Interior).

not clear that the Act establishes a sensible cost-benefit rule.[267] Fixing the flaws in FISMA's implementation, or at a minimum understanding those flaws better, is required before implementing any new legislation targeting government information security.

A second possible approach for promoting software security is for the government to apply legislation or regulations to its own agencies first. Then, these laws and regulations could be revisited as new knowledge of their likely costs and benefits emerges.

Finally, the government could work with other nations to strengthen the Common Criteria software security review process that many agencies already rely on. Common Criteria's reviews could assess the software design and development process, perhaps including several checkpoints along the development path, as opposed to a single review of final products. Another reform might be to shift Common Criteria's review toward a heavier emphasis on secure design and development procedures—for example, threat assessments and risk analysis—and away from end product documentation. The hurdles for this option are largely political. Nations at different points along the economic development curve would likely view Common Criteria modifications through different lenses, but given the benefits to be obtained from more secure software products, reform could be in all nations' best interests.

E. Voluntary Corporate Actions

Secure Design and Implementation: One obvious private sector action is an increased focus on security in operations. For software developers, that could include the improved product design and development procedures discussed above. It could also cover automated patching for those bugs or vulnerabilities that inevitably slip through. For internet access providers, better security in operations would imply improved network design to create barriers for unauthorized access. It could also include security monitoring procedures, such as automated virus scanning of email attachments. Care would need to be taken,

267. The Interior Secretary notes that she is "unclear on whether the agency should be spending money to correct defects that may pose minimal risk. In her letter to OMB she requests a clearer definition of 'adequate' security." *Id*.

however, to balance security benefits against any encroachments on customer privacy.

Bounties: A narrower, but still potentially fruitful avenue involves bounty hunting. Some software developers have voluntarily begun to assume a stronger role in curbing software security breaches by adopting "bounty" programs.[268] SCO, Microsoft, and others have used this approach previously.[269] A company may offer a bounty of $25,000 to every individual who provides information that leads to the arrest of a malicious hacker or who identifies a new security flaw in its programs. The appeal of this solution is that software developers can tailor the cost to the benefit, offering larger rewards for more serious problems and lower rewards for less serious ones. Programs of this sort also do not require much in the way of infrastructure. On the other hand, there is no guarantee that the tips received will lead to actual arrests or have any deterrent effect on malicious hackers. For example, Microsoft's bounty reward proved ineffective in efforts to catch the creators of the MS Blast worm and MyDoom virus,[270] although, the company's Sasser worm bounty was successful.[271] Nevertheless, because bounty programs cost little to establish, only pay out when a benefit is achieved, and can be set at a level that guarantees that benefits exceed any costs, this sort of tactic is recommended.

F. Cyber Insurance

Insurance can combine both the public and private sectors. For example, to reduce automobile accidents and provide for a compensation scheme when accidents do happen, the government requires all drivers to obtain insurance before they can drive. The rule is government enforced; the insurance is privately supplied.

---

268. Barnes would make such bounty programs mandatory. Barnes, *supra* note 7, at 322–24. Professor Larry Lessig is an advocate of this approach as well. *See* Lawrence Lessig, *Code Breaking: A Bounty on Spammers*, CIO INSIGHT, Sept. 2002, at 27–28, *available at* http://www.cioinsight.com/print_article/0,3663,a=31039,00.asp.

269. Dennis Fisher, *Microsoft Puts Bounty on Virus Writers*, EWEEK.COM, Nov. 5, 2003, http://www.eweek.com/article2/0,4149,1373578,00.asp; Ken Mingis, *SCO Sets Bounty for Worm Writer*, PCWORLD.COM, Jan. 27, 2004, http://www.pcworld.com/article/id,114479-page,1/article.html.

270. Robert Lemos, *Virus Writers Elude Microsoft's Bounty Hunt*, CNET NEWS.COM, Nov. 5, 2004, http://news.com.com/2100-7349_3-5439456.html.

271. *See* Banisar, *supra* note 221.

In recent years, cyber insurance has begun to emerge as a viable option for companies.[272] Cyber insurance policies cover damages caused by a full array of security problems, including viruses, worms, denial of service attacks, and data theft or corruption. Statistics on cyber insurance indicate that it is still uncommon, but most observers expect the number of policyholders to increase over time. Both the CSI/FBI 2005 survey and the Deloitte Touche Tohmatsu survey find that around 25% of respondents now have cyber insurance.[273] As of 2005, roughly fifteen companies offered cyber insurance.[274] Among those offering cyber security supplements are some of the largest insurance companies, including American International Group (AIG), Chubb, Hartford Insurance Group, Insuretrust, Lloyds of London, and Marsh & McLennan.

Companies offering cyber insurance typically categorize policyholders into risk groups. Insuretrust rates firms on a scale of one to thirty, for instance. Those firms with the best security practices, such as those with professional security systems installed, fall into the lowest risk groups and pay the lowest premiums.[275] Just as with other forms of insurance, cyber insurance could provide increased incentives to corporate end users, internet access providers, and software developers to increase IT security. This route has the advantage of being market driven, so the price of additional security will be weighed against the benefits that the added security has to offer. Minor security breaches and nuisances will be ignored, even as genuine security risks are focused on.

---

272. The idea of an insurance market specifically tailored to IT security has captured the imagination of a number of scholars, and the literature on the topic is considerable. *See, e.g.*, Joshua Gold, *Insurance Coverage for Internet and Computer Related Claims*, 19 COMPUTER & INTERNET L. 8 (2002); Anna Lee, *Why Traditional Insurance Policies Are Not Enough: The Nature of Potential E-Commerce Losses & Liabilities*, 3 VAND. J. ENT. L. & PRAC. 84 (2001); Jay P. Kesan et al., *The Economic Case for Cyberinsurance* (Univ. of Ill. Coll. of L., Working Paper No. 1001, 2004).

273. CSI/FBI 2005 SURVEY, *supra* note 84, at 10 fig.12; DELOITTE TOUCHE TOHMATSU, *supra* note 131, at 18.

274. Len Strazewski, *internet Security—A Growing Risk for Businesses*, ROUGH NOTES, Sept. 5, 2005, at 186, 189, *available at* http://www.roughnotes.com/rnmagazine/2005/september05/09p186.htm. The number of insurance companies offering cyber insurance has nearly doubled since 2003 when roughly eight companies were offering cyber insurance. *See* Mark Willoughby, *New Regulations Have Companies Turning to Risk Management*, COMPUTERWORLD, June 5, 2003, http://www.computerworld.com/securitytopics/security/story/0,10801,81827,00.html.

275. *See* Kesan et al., *supra* note 272, at 28.

As the cyber insurance market evolves, we should see a continued expansion of third party security raters helping to solve information discrepancies between software developers and insurers.[276] As noted above, BITS Financial Services Security Lab and ICSA Labs are two such ratings firms. If this industry were to develop further, as might be expected with greater use of cyber insurance, it could provide a non-regulatory solution to the information asymmetry problems discussed earlier.

We therefore see cyber insurance as an extremely promising route to solving the identified market failures in software system security. If the market were to develop sufficiently, it might be able to serve both consumers and businesses. Whether cyber insurance should be mandated at this point, however, is difficult to judge, again due to the limited data available.

V. CONCLUSION

This Article has evaluated the law and economics of software system security. The economics are fairly simple, the law less so. Both inform our evaluation of proposed policy changes. The Article has tried to take a comprehensive approach in its analysis, considering a wide range of security issues and evaluating the case for government intervention from a number of angles. The analysis offers several insights.

First, software security problems vary considerably. For example, some breaches are relatively harmless while others have the potential for significant harm. Different types of security issues suggest different types of solutions. No single policy is likely to effectively address all software system security problems. Moreover, broad interventionist proposals are difficult to justify.

Second, based on the available data, it is not clear that each aspect of software security poses a particularly large problem in terms of the damages inflicted by a breach. Anecdotal evidence suggests that security attacks lead to a wide range of damages. The size of the problem is a crucial aspect of deter-

276. Underwriters' Laboratories, for example, grew out of the insurance industry's need to accurately assess the fire hazards associated with the proliferation of electrical appliances in the mid to late 1800s. *See* Harry Chase Brearly, *A Symbol of Safety: The Origins of Underwriters' Laboratories*, *in* REPUTATION: STUDIES IN THE VOLUNTARY ELICITATION OF GOOD CONDUCT 78 (Daniel B. Klein ed., 1997).

mining whether government intervention is justified. For example, media reports of worms and viruses are plentiful, but the economic impact of malware of this type appears to cover a broad range—from relatively minor system slowdowns to millions of dollars of damage in corrupted data. Better data on the damage done by attacks is a key to any intelligent policy plan. One clear path for policymakers is to support the collection of more systematic data on the costs imposed by software system security failures. In order to ensure broad participation, those data collection efforts should strive to maintain victims' anonymity.

Third, market failures in the provision of security are evident, but do not appear to apply to all parties equally. A case can be made for at least two distinct market failures: information asymmetries and security externalities. The first is likely to be an issue for end consumers and small businesses, but less so for larger enterprises. The second is likely to be an issue for some consumers and businesses, but evidence of self-selection into stronger security regimes suggests that the negative effects of software security externalities can be mitigated. Given the evolving state of software security issues, especially those related to the internet, which is still relatively young, it may be that it is simply too soon to tell whether the market has failed irrevocably or whether it is working reasonably well. Some innovative market-based solutions have already emerged that address a number of software security problems. Whether the market response and existing law are sufficient to address all important security issues is still an open question.

Unfortunately, many existing laws targeting software security are likely to be ineffective tools for correcting the identified market failures. Most existing legislation focuses on criminalizing cyber infractions, but relatively few cyber criminals are caught and prosecuted. Of course, malicious actions should remain illegal. These laws, however, are unlikely to deter cyber criminals given the anonymous and flexible nature of online crime. The three laws, discussed above, that focused on research, mandatory reporting for serious breaches, and security requirements for government agencies, are the exceptions. These laws address areas where government intervention at least has a chance of providing net benefits, assuming the laws can be implemented effectively. In particular, anonymous security breach reporting could help to improve information on security issues and provide stronger incentives for individual

companies to invest in securing their systems. Care needs to be taken in designing the exact reporting rules so that disclosures do not encourage copycat attacks or have other unintended consequences.

Finally, although several proposals for addressing security may be worth considering, most would require modification to ensure that they do more good than harm. For example, a lemons law for software would likely increase incentives for secure software among developers, but would also reduce incentives for end users (corporate and consumer) to take adequate security precautions. Implementing heavy-handed measures could reduce innovation among developers as well as reduce security investment among users below some optimal level. A modified version, in which software developers could be held liable for the damages caused only from the time of the first security breach to the time the first security patch was released, could merit consideration. Even here though, market based solutions—such as better software design and development, automated software patches, new security products, and corporate bounty programs—are emerging that could make such a law unnecessary. Moreover, unintended consequences—such as the issuing of patches prematurely solely to limit the liability period—could reduce the potential benefits. This option would therefore need to be analyzed carefully before deciding on any policy. Other possibilities, such as the developing cyber insurance market, appear to be promising.

The government could take some steps to improve software system security that would not require private sector regulation. For instance, it could focus on making the FISMA work more effectively so that agencies improve their security. The government might also work with other nations to improve the existing Common Criteria security review process for software products.

The best step policymakers could take immediately would be to encourage reporting of security breaches. This could entail establishing a government-led system for anonymous reporting, or funding smaller scale research projects that examine the actual costs involved in different kinds of breaches. A more dramatic step in this direction would be to expand California's reporting law to the federal level. Whatever is done, it should be done carefully, with an eye toward minimizing the overall social costs associated with software security problems.