Stanford Technology Law Review

VIRUS EX MACHINA

RES IPSA LOQUITUR

MEIRING DE VILLIERS[*]

CITE AS: 2003 STAN. TECH. L. REV. 1
http://stlr.stanford.edu/STLR/Articles/03_STLR_1

I. INTRODUCTION

¶1 The global presence, explosive growth and open access of the Internet and modern communications technology have dramatically increased the vulnerability of a provider or distributor of software to liability for harm caused by errors, logical flaws and other factors that may cause a computer system error. These hazards include the threat of malevolent software and rogue programs, such as computer viruses, that are capable of spreading rapidly and causing widespread and substantial damage to data and programs.[1]

¶2 A web site controller, for instance, may face liability for a Java applet on her home page which deletes data on a particular type of browsing computer. The system operator in a workplace who becomes aware that an internal network is infected with a virus may have a duty to external e-mail recipients not to spread infected material, either by informing employees, blocking all external e-mail traffic or including warnings with outgoing e-mail.[2] Bulletin boards, which allow downloading and uploading of software, are particularly vulnerable to computer virus infection due to the sheer quantity of transactions performed through bulletin board systems.[3]

[*] Dept. of Management Science & Engineering, Stanford University, Stanford, CA 94305-4023, mdv@stanford.edu, (650) 725-8214. Dedicated to My Latin Teachers. Special thanks to the editors of The Stanford Technology Law Review for superb editing. I am grateful to Ian Ayres, Ed Felten, Peter Glynn, Sy Goodman, Mark Grady, Gregory Grove, Keith Hylton, David Kaye, Lawrence Lessig, Charles Nesson, Mitchell Polinsky, Robert Rabin, Kevin Soo Hoo, and Robert Weisberg, for useful comments. I am also pleased to acknowledge valuable feedback and stimulating discussions from participants at the following conferences: The Fifth Annual International Conference on Industrial Engineering Theory, Applications and Practice, at National Tsing Hua University, Hsinchu, Taiwan (Dec. 2000), Eleventh Annual Meeting of The American Law and Economics Association, Georgetown University Law Center, Washington, D.C. (May 2001), The Eleventh Annual Virus Bulletin Conference, Prague, Czech Republic, (September 2001), and The Center for Internet and Society Seminar, Stanford Law School, (April 2002). Financial support from the U.S. Department of Defense and the John M. Olin Foundation, at Stanford Law School, is gratefully acknowledged.

[1] KEN DUNHAM, BIGELOW'S VIRUS TROUBLESHOOTING POCKET REFERENCE xix-xxiii (2000); Jeffrey O. Kephart et al., *Blueprint for a Computer Immune System*, IBM THOMAS J. WATSON RES. CENTER REP., *available at* http://www.research.ibm.com/antivirus/SciPapers/Kephart/VB97/ ("There is legitimate concern that, within the next few years, the Internet will provide a fertile medium for new breeds of computer viruses capable of spreading orders of magnitude faster than today's viruses... [T]he explosive growth of the Internet and the rapid emergence of applications that disregard the traditional boundaries between computers threaten to increase the global spread rate of computer viruses by several orders of magnitude."); PHILIP FITES ET AL., THE COMPUTER VIRUS CRISIS 21 (2d ed. 1992); Carey Nachenberg, *Future Imperfect*, VIRUS BULL. (1997) ("With the ubiquitous nature of the Internet, new viruses can be made widely accessible within minutes.").

[2] CLIVE GRINGRAS, THE LAWS OF THE INTERNET 61-62 (1997), 61, 62. An English court held that a defendant who stored biological viruses had a duty to cattle owners who would be affected by the spread of the virus. Weller and Co. v. Foot and Mouth Disease Research Institute [1965] 3 All ER 560, at 570. ["[T]he defendant's duty to take care to avoid the escape of the virus was due to the foreseeable fact that the virus might infect cattle in the neighborhood and cause them to die. The duty is accordingly owed to the owners of cattle in the neighborhood . . . ."].

[3] FITES, *supra* note 1, at 60.

¶3 A victim of a virus attack may bring legal action under a negligence theory against the provider of the infected software, as well as against entities involved in its distribution, such as web site operators. Negligence is a breach of the duty not to impose an unreasonable risk on society.[4] To pursue a negligence cause of action, a victim of viral infection would have to prove (1) that the defendant had a duty to the plaintiff to take reasonable care to avoid infection of the software distributed through her channels, (2) that she breached that duty, and (3) that the breach was the actual and legal cause of the plaintiff's loss. The basic duty test considers whether the defendant created the risk that caused the loss,[5] but the courts also impose a duty of due care, for instance, on one who possesses a "special relationship" with the plaintiff.[6] Given the public awareness and publicity surrounding virus attacks and computer security more generally, courts are likely to find that software providers and distributors generally do have a duty not to impose an unreasonable risk of viral infection on those foreseeably affected.[7] A software provider, for instance, who invites customers to download her product from the Internet creates a risk that she may transfer a virus along with her software. Everyone who downloads the software is within the scope of the risk and may have a cause of action if harmed by a virus.

¶4 Negligence has to be proved, and will not be presumed merely because an injury has occurred. The injury may, after all, be due to a legally "unavoidable" error, i.e., an error that occurred in spite of reasonable precautions.[8] What is required is evidence from which reasonable persons may conclude that there is a greater than 50 percent probability that the injury was caused by defendant's negligence.

¶5 Breach of duty in a negligence action is typically proven by identifying an untaken precaution, and by showing that the untaken precaution would have yielded greater benefits in accident reduction than its cost.[9] Quantitative models of the costs and benefits of antiviral defenses have appeared in computer security literature, making evaluation of the efficiency of an untaken anti-viral precaution feasible, at least in principle.[10] However, in cases involving complex and novel virus strains, and where lapses in anti-viral precautions leave no evidentiary trace, such direct proof may be impossible. Verification of a provider's compliance with the non-durable, repetitive component of virus detection is sometimes difficult or impossible. In such cases, where direct proof of negligence is not feasible, a plaintiff would have to rely on circumstantial evidence in order to recover damages.

[4] PROSSER AND KEETON ON THE LAW OF TORTS (5th ed. 1984), § 31. RESTATEMENT (SECOND) OF TORTS § 282 (describing negligence as conduct "which falls below the standard established by law for the protection of others against unreasonable risk of harm.").

[5] Weirum v. RKO Gen., Inc., 15 Cal. 3d 40, 46 (Cal. 1975).

[6] Lopez v. Southern Cal. Rapid Transit Dist., 710 P.2d 907, 911 (Cal. 1985); *see also*, Tarasoff v. Regents of Univ. of Cal., 551 P.2d 334, 342 (Cal. 1976).

[7] FITES, *supra* note 1, at 141-142 (explaining that Bulletin Board System operators provide a forum for exchange of information, data and software. Thus, a BBS operator may have a duty to screen uploaded software for malicious components or to at least warn users to use caution in using downloaded software); *see also*, David L. Gripman, *The Doors are Locked but the Thieves and Vandals are Still Getting In: A Proposal in Tort to Alleviate Corporate America's Cyber-Crime Problem*, 16 J. MARSHALL J. COMPUTER & INFO. L. 167, 170 (1997) (asserting that in the context of electronic commerce, a computer user can be considered to have the requisite relationship of proximity with any other computer user with whom she is in contact to establish a duty of care); Palsgraf v. Long Island R.R. Co., 248 N.Y. 339 (N.Y. 1928) (establishing the precedent that a duty extended only to those foreseeably affected.).

[8] DAVID BENDER, COMPUTER LAW: EVIDENCE AND PROCEDURE 8-42 (1982); CHIN-KUEI CHO, AN INTRODUCTION TO SOFTWARE QUALITY CONTROL 4, 12-13 (1980) (stipulating that a software provider is under a duty to invest resources in program debugging only up to the point where the cost of additional debugging would outweigh the benefits of further error reduction.); Thomas G. Wolpert, *Product Liability and Software Implicated in Personal Injury*, DEF. COUNS. J.519, 523 (1993) ("By the time a product is completely debugged, or nearly so, most likely it is obsolete.").

[9] Mark F. Grady, *Untaken Precautions*, 18 J. LEGAL STUD. 139, *passim* (1989); Delisi v. St. Luke's Episcopal-Presbyterian Hosp., 701 S.W.2d 170 (Mo. App. 1985) (Plaintiff patient had to specify appropriate treatment he should have been given); Tower v. Humboldt Transit Co., 169 P 227 (1917) (Negligence must be proven and will not be presumed; negligence is failure to exercise due care, i.e. degree of care demanded by circumstances; plaintiff specified precautions that defendant transit company had failed to undertake.).

[10] Fred Cohen, *A Cost Analysis of Typical Computer Viruses and Defenses*, 10 COMPUTERS & SECURITY 239-250 (1991); *see also*, FREDERICK B. COHEN, A SHORT COURSE ON COMPUTER VIRUSES 117-118 (2d ed. 1994) (discussing limited availability and quality of statistical data.).

¶6 The evidentiary difficulties originate from the nature of virus and virus detection technology. Antivirus precautions consist of a durable as well as non-durable component. A durable precaution typically has a long service life once it is installed. Use of a durable precaution must usually be complemented by shorter-lived, non-durable precautions. A medical example illustrates the distinction between durable and non-durable precautions. A kidney dialysis machine is a typical durable precaution. A dialysis machine has a long service life once it is installed, but it cannot function properly without complementary non-durable precautions, such as regular monitoring of the hemodialytic solution.[11]

¶7 Analogous to the medical example, computer virus detection and elimination requires durable precautions, such as a virus scanner and signature database, complemented by non-durable precautions, such as regularly updating the signature database and monitoring the output of the scanner.[12] A scanner reads software code and searches for patterns that match the known viral patterns in its database.

¶8 Compliance with durable precautions can be verified more easily than compliance with non-durable precautions. While it can in principle be verified whether the defendant had a scanner or not, it may be considerably more difficult to prove improper maintenance of the signature database. Use of a virus scanner must be complemented by updating the signature database at an optimal (due care) rate, say, once at the beginning of each workday. A virus may be transmitted because the database was not properly updated, e.g., the software provider skipped an update. The virus may also have been transmitted because the signature of the strain had not yet been commercialized at the time the software product was scanned, hence unavailable for inclusion in the database. The former situation would constitute a breach of duty, and thus negligence, while the latter would not. However, it may be impossible to distinguish between negligent and non-negligent virus transmission, without evidence of the precise identity of the virus strain and the exact time the software was scanned. At the time a plaintiff discovers the virus, the defendant's database may have been fully updated. Proving that it was not updated at the time the software was scanned may be impossible, especially in the case of a recently-commercialized, novel virus strain. Investment in a scanner, and subscription to and delivery of signature updates, all leave a paper trail which may help prove negligence in court. Other non-durable precautions, such as faithfully implementing each update and paying attention to all alerts, including false alarms, are sometimes harder to verify.

¶9 The evidentiary problem is compounded by the growing sophistication of viruses as well as virus detection technology.

¶10 Earlier viruses used encryption to scramble their signature, making it unrecognizable to a virus scanner. The encrypted virus consisted of a decryption routine and an encrypted virus body. When such a virus gained control of the computer, the decryption routine executed first and decrypted the virus body. The decrypted virus then executed and infected programs and files by making a copy of the decrypted virus body and its decryption routine, encrypting the copy and attaching both to the new host. The virus was programmed to change the encryption key from infection to infection, making it hard for a scanner to recognize its signature. In simple encrypted viruses of this kind, the decryption routine remained constant, a weakness that has been exploited by anti-virus researchers who developed scanners capable of recognizing byte sequences characterizing specific decryption routines.

¶11 Virus authors responded by creating polymorphic viruses which contain a third component, namely a mutation engine that generates decryption routines that change randomly with each new infection. The mutation engine and virus body are both encrypted. When control is transferred to the virus, the decryption routine executes first and decrypts both the virus body and the mutation engine. Control is then transferred to the virus body. The virus body executes and makes a copy of

[11] Mark F. Grady, *Why Are People Negligent? Technology, Nondurable Precautions, and the Medical Malpractice Explosion*, 82 NW. U.L. REV. 293, 299 (1988).

[12] See Section II of this article for a review of virus detection technologies.

itself as well as the mutation engine in the computer's RAM. The mutation engine then generates a random new decryption routine that bears little if any resemblance to the prior decryption routines, yet is capable of decrypting the virus. The virus then encrypts the newly created copy of the virus body and the mutation engine. It infects a new host by appending the new encryption routine and the newly encrypted virus body and mutation engine to the host. The result is a virus body with a signature and decryption routine which both vary randomly from infection to infection, so that no two infections "look alike." [13]

¶12 Polymorphic virus technology therefore may create evidentiary challenges. The transmission of such a virus by a software or service provider may be attributable to negligence on the part of the provider. The provider would be considered negligent if she had failed to detect a virus containing features such as a well-known signature or decryption routine. It is also possible that even reasonable precautions would have failed to detect the virus. The virus may, for instance, have carried a complex signature and/or decryption routine at the time of quality control. The signature may, for instance, not yet have been commercialized, and hence been unavailable for inclusion in the signature database of a virus scanner. Because of the random evolution of the signature and decryption routine, the nature of its signature and the features of its decryption routine at the time of scanning (quality control) would be unverifiable, making direct proof of negligence impossible.

¶13 An evidentiary problem may also arise where a virus is programmed to remain dormant and execute at random times. A software provider may use a virus detection system, such as an activity monitor or integrity checker, that detects the presence of a virus only after execution. Thus, a virus may be transmitted undetected in a software product, in spite of diligent use of the detection system, simply because it had not yet executed at the time the software was released. If the transmitted virus subsequently executes and causes harm, the plaintiff faces an evidentiary dilemma: was the virus transmitted because of the defendant's negligence, (e.g. failure to heed an alert), or did the virus escape detection because it had not yet executed at the time of the software's release in which it was embedded? The random execution trigger makes this distinction, and direct proof of defendant's negligence, impossible ex-post.[14]

¶14 In summary, direct proof of negligence in a case involving virus infection is complicated by the dynamic nature of viruses and virus detection technology, and the frequent absence of an evidentiary trace of failure to comply with non-durable precautions. In some cases, such as those involving a well-known virus that could have been eliminated efficiently, direct proof of negligence is feasible. In cases involving novel and complex viruses, direct proof is sometimes impossible. This article develops a theory of circumstantial evidence, aimed at easing the evidentiary burden of victims of viral infection. The theory would allow a virus victim to establish a cause of action without specific proof of negligence.

¶15 Negligence, like any other fact, can be proved by indirect, or circumstantial, evidence. The tort doctrine *res ipsa loquitur* is a potentially powerful doctrine of circumstantial evidence. Translated from Latin, the phrase means "the thing speaks for itself," referring to the legal inference that a harmful event may be of such a nature that its *mere occurrence* and the circumstances surrounding it may permit an inference that a defendant was negligent. *Res ipsa* is a legal rule that allows plaintiffs to establish negligence without having to prove *specific* negligence, to plead a specific untaken precaution, or to

[13] *See, e.g.*, Carey Nachenberg, *Understanding and Managing Polymorphic Viruses*, THE SYMANTEC ENTERPRISE PAPERS, Volume XXX.

[14] Mission-critical software presents a further illustration of evidentiary difficulties associated with virus infection. Mission-critical software may require an exceptionally high duty of due care, such as use of heuristic scanning technology capable of detecting unknown virus strains. Such detection systems are often based on artificial intelligence methods and neural learning. The detection system's capability to detect unknown viruses expands as learning progresses. Suppose a virus evades detection and causes considerable harm. The virus may have been transmitted because the detection system was simply not capable of capturing that particular strain, at the time of release, i.e. the learning process had not yet advanced sufficiently. On the other hand, transmission may have been due to the defendant's non-compliance with a non-durable precaution, such as failing to pay attention to an alert. Verification of the defendant's culpability would require identification of the virus strain and reconstructing the learning stage of the system, as well as possible behavior exhibited by the virus, at the time the infected software was released, perhaps an impossible task.

present clear evidence of what went wrong, as long as the harmful event and the circumstances surrounding it "speak of the defendant's negligence."

¶16 The *res ipsa* inference can be weak or strong, depending on the circumstances of the case.[15] A generic computer malfunction, for instance, does not present a strong *res ipsa* case.[16] Programmer negligence cannot be inferred merely because a computer has functioned improperly. There are many factors beyond the control of the programmer, such as power failure, hardware error and system program error, which could interfere with an application program and cause it to fail. Furthermore, these factors are often transient and difficult to detect after the fact.

¶17 This article will argue that although damage resulting from a general computer malfunction presents a weak *res ipsa* case, damage due to a computer virus attack constitutes a strong *res ipsa* case. This conclusion follows from an analysis based on the computer science of the structure, operation and detection of computer viruses, the law and economics of virus prevention, and a probabilistic analysis of the *res ipsa* inference of negligence.

¶18 A probabilistic analysis derives a mathematical formulation of the *res ipsa* conditions and identifies the factors that make a strong *res ipsa* case, namely (i) a high avoidable to unavoidable error ratio, and (ii) a high *a priori* probability of negligent causality. An accident has a high avoidable-to-unavoidable error ratio if a large proportion of the factors that may cause the accident are avoidable by due care. A plane crash, for instance, has a high avoidable-to-unavoidable error ratio because the availability of sophisticated technology, such as computerized navigation equipment, coupled with a legal duty to use it, significantly reduce the likelihood of a crash.

¶19 An economic analysis suggests that virus infection presents a strong *res ipsa* case based on the following factors. First, virus prevention has a high avoidable-to-unavoidable error ratio, analogous to the use of the modern airplane. Advances in anti-virus research have yielded sophisticated and effective anti-viral technology, capable of detecting and eliminating a substantial proportion of even unknown virus strains.[17] Furthermore, the high danger rate associated with viral infection and the modest cost of precautions create a legal duty to implement intense anti-viral precautions. This is especially true where the risk of infection threatens mission-critical software in the so-called "national critical information infrastructure," where the danger rate is particularly high.

¶20 Second, virus detection technology, including digital signature analysis, can be helpful to a plaintiff building a *res ipsa* case. A computer malfunction, such as a system crash, may be due to a multitude of factors, including virus infection.[18] Identifying a virus as the culprit by its digital signature and side-effects strengthens the *res ipsa* inference by transforming the computer error from a case with a low avoidable to unavoidable error ratio (generic computer malfunction) to one with a high avoidable to unavoidable error ratio (malfunction due to virus).

¶21 Third, an economic analysis of virus detection technology shows that software providers have an incentive to invest in anti-viral precautions at a level that maximizes expected profitability, but that falls below the legally required level of due care. The larger this gap, the greater the likelihood that an avoidable virus strain will be transmitted. The result is a high *a priori* probability of negligent virus transmission. All three factors suggest that a case involving a virus attack will generally be a strong *res ipsa* case.

¶22 This article is organized as follows: Section II discusses the operation and structure of computer viruses, including technical anti-viral defenses. Section III introduces the legal definition

[15] Mark F. Grady, *Res ipsa Loquitur and Compliance Error*, 142 U. PENN. L. REV. 887, 912 n. 81 ("Res ipsa creates an inference of negligence. Res ipsa cases are strong or weak, judged by how clearly the plaintiff's proof and the other circumstances suggest that the defendant's negligence was the culprit.").

[16] Daniel J. Hanson, *Easing Plaintiffs' Burden of Proving Negligence for Computer Malfunction*, 69 IOWA LAW REVIEW 241, 252-253 (1983).

[17] *See, e.g.*, Carey Nachenberg, *Future Imperfect*, VIRUS BULLETIN, Aug. 1997, 6, 7 (illustrating that state-of-the-art technology is capable, for instance, of detecting up to 80 percent of unknown strains).

[18] Jeffrey O. Kephart et al., *Fighting Computer Viruses*, SCIENTIFIC AMERICAN, Nov. 1997, at 91 (explaining that Windows tends to crash in the presence of certain kinds of viruses).

of *res ipsa loquitur* and presents a probabilistic analysis of the *res ipsa* inference. The probabilistic model identifies and quantifies the factors that support an inference of negligence, and thus make a strong *res ipsa* case. Sections IV and V analyze computer virus infection as a *res ipsa* case, concluding that virus infection constitutes a strong *res ipsa* case. Section VI addresses aspects of damages, including a model of damages and analysis of the economic loss rule in a computer virus context. A final section discusses the implications of major results of the article.

II. OPERATION AND STRUCTURE OF COMPUTER VIRUSES

¶23 Malevolent software is software intended to damage or disrupt the operation of a computer system. The most common of these rogue programs, and the focus of this paper, is the computer virus. Other forms of malicious software include so-called logic bombs, worms, Trojan horses, and trap doors.[19]

¶24 The term "virus," Latin for "poison," was first formally defined in the context of computer programs by Dr. Fred Cohen in 1983,[20] though the concept goes back to John von Neumann's studies of self-replicating mathematical automata in the 1940s.[21] In his Ph.D. dissertation, Cohen defined a virus as any program capable of self-reproduction. This simple definition is overly general, as it includes programs such as compilers and editors.[22] A more realistic definition would describe a computer virus as a series of instructions, (i.e. a program), that (i) infects other computer programs and systems by attaching itself to a host program in the target system, (ii) executes when the host program is executed, and (iii) spreads by cloning itself, or part of itself, and attaching copies to other host programs on the system or network.[23] In addition to self-replicating code, viruses also contain code capable of causing side-effects, such as the destruction or corruption of data.

¶25 Viruses spread by finding and infecting new host computers and programs.[24] Any file that contains executable code is a potential virus carrier, but the three most common target "host" areas in a computer system are the system boot code,[25] the operating system, [26] and application programs, such as word processing, spreadsheet or communication programs.[27]

¶26 Viruses can spread via a variety of entry points. Common routes of virus infection include pirated software, software downloaded from bulletin boards, shareware, and public domain software.[28] Viruses often spread through the exchange of infected floppy disks. If, for instance, a non-infected computer disk were inserted into the drive of a virus-infected computer, the resident virus could replicate and spread by copying itself onto the "clean" disk. Were this newly-infected disk inserted into a different computer, the virus would typically repeat the replication and spreading

[19] *See, e.g.*, Eugene H. Spafford, *Computer Viruses.* INTERNET BESIEGED 75-78 (Dorothy E. Denning & Peter J. Denning eds., 1998).

[20] Fred Cohen, Computer Viruses (1985) (unpublished Ph.D. dissertation, University of Southern California) (on file with author).

[21] Kephart et al., *supra* note 18, at 88. Dr. Gregory Benford first published the idea of a computer virus as "unwanted code." Benford wrote about actual "viral" code, capable of replication, in conjunction with a "vaccine" program to defeat it. Spafford, *supra* note 19, at 74.

[22] Spafford, *supra* note 19, at 75.

[23] JOHN MACAFEE & COLIN HAYNES, COMPUTER VIRUSES, WORMS, DATA DIDLERS, KILLER PROGRAMS, AND OTHER THREATS TO YOUR SYSTEM 1 (1989); COHEN, A SHORT COURSE ON COMPUTER VIRUSES, *supra* note 10, at 1-2.

[24] MACAFEE & HAYNES, *supra* note 23, at 1; Myron L. Cramer & Stephen R. Pratt, *Computer Virus Countermeasures – A New Type of Electronic Warfare.* ROGUE PROGRAMS: VIRUSES, WORMS, TROJAN HORSES 247 (Lance Hoffman ed., 1990) ("The ability to propagate is essential to a virus program."); Spafford, *supra* note 19, at 73-75.

[25] The boot sector executes the start-up procedures of the computer system, such as installing the operating system. MACAFEE & HAYNES, *supra* note 23, at 61.

[26] The operating system is a program that manages computing resources and performs basic housekeeping functions in a computer system, such as allowing user tasks to interface with machine hardware, controlling inputs and outputs, and running application programs. *Id.* at 63.

[27] *Id.* at 63, 71.

[28] JAN HRUSKA, COMPUTER VIRUSES AND ANTI-VIRUS WARFARE 26-27 (1990).

process.[29] Currently, the most prevalent threats are macro viruses, which infect e-mail attachments, word processing documents and spreadsheets.

¶27 There are three mechanisms through which a virus can infect a program. A virus may attach itself to its host as a shell, an add-on, or as intrusive code.[30] A shell virus forms a shell around the host code, so that the latter effectively becomes an internal subroutine of the virus. The host program is replaced by a functionally equivalent program that includes the virus, where the virus executes first, then allows the host code to begin executing. Boot program viruses are typically shell viruses. In contrast, most viruses are of the add-on variety. They become part of the host, by appending their code to the host code without altering it. The viral code alters the order of execution, executing itself first and then the host code. Macro viruses are typically add-on viruses. Finally, an intrusive virus overwrites some or all of the host code, replacing it with its own code.

¶28 Viruses can also be classified according to the locations they infect: file infectors, boot sector viruses, and macro viruses.[31] File infectors, as the name suggests, infect files containing applications, such as spreadsheet programs or games. Boot sector viruses reside in the boot sector, namely the program code that loads the rest of the computer's operating system. Macro viruses infect the macro commands ("macros") of files that are usually regarded as data, rather than programs. Macros are programs that perform functions such as opening data files and performing spreadsheet calculations. Although pure data cannot be infected, macros embedded in data files may be infected by a virus. Text files may also contain macro editor commands that are executed when the file is read by the editor. Such editor commands may likewise be infected. Macro viruses spread particularly rapidly, as many people share data files. For example, the first macro virus "observed in the wild," (which was named "Concept" and initially targeted to infect Microsoft Word documents), was once the most prevalent virus in the world.[32]

¶29 Viruses have two major components: a module that manages its replication and spreading mechanisms as well as a payload module that manages side-effects. The side-effects range from relatively harmless, (and even humorous), to highly destructive. A malignant virus has the ability to harm a computer system by changing or destroying data, such as information in spreadsheets, word processing documents and data bases.[33] In addition to deleting data or system files, it may make more subtle but equally pernicious changes, such as transposing numbers or moving decimal places.[34] In contrast, benign viruses do not intend to destroy data or programs but rather do relatively harmless things like displaying a message or image on the user's screen. Benign viruses can still be disruptive and expensive, however, by consuming valuable computing resources.[35]

A. Technical anti-virus defenses

¶30 Technical defenses against viral infection come in four broad categories: activity monitors, scanners, integrity checkers, and heuristic techniques.[36]

[29] MACAFEE & HAYNES, *supra* note 23, at 1.

[30] Spafford, *supra* note 19, at 79; FRITES ET AL., *supra* note 1, at 73-75.

[31] Kephart et al., *supra* note 18, at 88.

[32] *Id.* at 89.

[33] HRUSKA, *supra* note 28, at 17-18. In addition to self-replicating code, viruses often also contain a payload. The payload is capable of producing malicious side-effects. *See* COHEN, A SHORT COURSE ON COMPUTER VIRUSES, *supra* note 10, at 8-15 (examples of malignant viruses and what they do).

[34] MACAFEE & HAYNES, *supra* note 23, at 60-61.

[35] Viruses can cause economic losses, e.g. by filling up available memory space, slowing down the execution of important programs, locking keyboards, adding messages to printer output, and effectively disabling a computer system by altering its boot sector. *See, e.g.*, FRITES ET AL., *supra* note 1, at 23-26 (noting that the IBM Christmas card virus stopped a major international mail system just by filling up all available storage capacity). *See* Section VI, *infra,* for an analysis of damages from computer virus infection. *See also* COHEN, A SHORT COURSE ON COMPUTER VIRUSES, *supra* note 10, at 15-21 (examples of benign viruses and how they operate).

[36] *See, e.g.*, Spafford, *supra* note 19, at 90-93; DUNHAM, *supra* note 1, at 78-83, 102-108.

¶31 *Activity monitors* are resident programs on the system that monitor activities, (i.e. commands the computer is requested to execute), and take action in response to suspicious events, such as attempts to rewrite the boot sector or to modify parts of main memory. When the monitor senses something suspicious, it may either halt execution and issue a warning to alert the user to the suspicious activity or simply take action to neutralize the activity. The weakness of this defense is that a virus may become activated before the monitor code, and escape detection until after execution. A virus may also be programmed to alter monitor code on machines like personal computers, which are not well-protected against such modifications. A further disadvantage of activity monitors is the lack of unambiguous and foolproof rules governing what constitutes "suspicious" activity. This may result in false alarms when legitimate activities resemble virus-like behavior. Recurrent false alarms may ultimately lead users to ignore warnings from the monitor. Conversely, not all "illegitimate" activity may be recognized as such, leading to false negatives.[37]

¶32 *Scanners* are the most widely used anti-virus defense. A scanner reads executables, including the operating system and boot sector, and searches for known virus patterns or so-called "signatures." The scanner announces any match with its database of known viral signatures as a possible virus. Scanners that employ algorithmic or heuristic checking may even detect polymorphic viruses, which complicate detection by changing their signatures from infection to infection.[38] The major advantage of scanners is that they are relatively easy to use. The major disadvantage (and limitation) of scanners is that they can only detect patterns contained in their database of known virus signatures. Keeping such a database updated, especially in an environment where new viruses appear rapidly, is burdensome. The use of scanners as an anti-viral defense is further complicated by the occurrence of false positives, when innocent data is identified as viral. A viral pattern in the database may match code that is actually a harmless component of otherwise legitimate data. A short and simple pattern will be found too often in innocent software, and produce many false positives. Viruses with longer and more complex patterns will give a false positive less often, but at the expense of more false negatives.[39]

¶33 An *integrity checker* is a program that generates a code, known as a "checksum," for files that are to be protected from viral infection. A checksum for a file may, for instance, be calculated using numerical values such as the total number of bytes in the file, the numerical value of the file size, and the creation date. The checksum effectively operates as a "signature" of the file. These check codes are periodically recomputed and compared to the original checksum. Tampering with a file will change its checksum. Hence, if the recomputed values do not match the original checksum, the file has presumably been modified since the previous check, and a warning is issued. Because viruses modify files by changing their contents when infecting them, a change in the checksum may be a sign of viral infection.[40]

¶34 The advantage of integrity checking is that it detects most instances of viral infection, because an infection must alter the file being infected. The main drawback is that it tends to generate many false alarms, as a file could change for many "legitimate" reasons unrelated to virus infection.[41] On some systems, files change whenever they are executed. A relatively large number of false alarms may trigger compliance lapses, as users may choose to ignore warnings or simply not use an integrity checker at all.

¶35 A fourth category of virus detectors uses *heuristic* detection methods. Heuristic rules are rules that solve complex problems quickly, but sub-optimally. Virus detection is a complex problem which is amenable to heuristic solution. It has been proven mathematically that it is impossible to write a program capable of determining with 100 percent accuracy whether a particular program is

[37] HRUSKA, *supra* note 28, at 75.

[38] Polymorphic viruses have the ability to "mutate" by varying the code sequences written to target files. To detect such viruses requires a more complex algorithm than simple pattern matching. *See, e.g.*, Spafford, *supra* note 19, at 89.

[39] DUNHAM, *supra* note 1, at 78-83; Kephart et al., *supra* note 18 at 89-90.

[40] FRITES ET AL., *supra* note 1, Figures 5.2-5.5, at 69-76; DUNHAM, *supra* note 1, at 79.

[41] DUNHAM, *supra* note 1, at 79; FRITES ET AL., *supra* note 1, at 125.

infected with a virus, from the set of all possible viruses, known as well as unknown.[42] Heuristic virus detection methods accept this limitation and attempt to achieve a solution, (detection rate), which is "pretty good," albeit less than perfect.

¶36 Heuristic virus detection methods look for "virus-like" behavior, including virus-like code structure and dynamic behavior. Heuristic anti-virus programs may, for instance, scan executable code to examine its structure, logic and instructions. Based on this examination, the program makes an assessment of the likelihood that the scrutinized program is a virus. The assessment is necessarily less than perfect and occasionally provides false positives and negatives. Nevertheless, state-of-the-art heuristic scanners typically achieve a 70-80 percent success rate at detecting *unknown* viruses.[43]

¶37 A major advantage of heuristic scanning is that it can detect viruses before they execute themselves and cause damage. Other generic anti-virus technologies, such as behavior monitoring and integrity checking, can only detect and eliminate a virus after exhibition of suspicious behavior, usually after execution. Heuristic scanning is also capable of detecting novel and unknown virus strains, the signatures of which have not yet been catalogued. Such strains cannot be detected by conventional scanners, which can only recognize known signatures.

¶38 A heuristic scanner typically operates in two phases. The algorithm first narrows the search by determining the most likely location to find a virus. It then analyzes the code from that location to determine its likely behavior upon execution. A static heuristic scanner, for instance, compares the code from the "most likely" location to a database of byte sequences commonly associated with virus-like behavior.[44]

¶39 A dynamic heuristic scanner uses CPU emulation to predict whether code is viral. It typically loads suspect code into a virtual computer, simulates its execution and observes its behavior. Because it is only a virtual computer, virus-like behavior can safely be observed in what is essentially a laboratory setting, with no need to be concerned about any real damage. Suspicious behavior, for instance, the capability to send e-mail with an attachment to everyone in an address book, is an indication of viral behavior.

¶40 Although dynamic heuristics can be time-consuming, due to the relatively slow CPU emulation process, they are sometimes superior to static heuristic scanners. This will be the case whenever the suspect code is (i) obscure and not easily recognizable as viral in its static state, but (ii) easily recognizable as viral in its dynamic state.

¶41 The explosive growth in new virus strains has made reliable detection and identification of individual strains very costly, while making heuristic methods of detection more important and increasingly prevalent.[45] Commercial heuristic scanners include IBM's AntiVirus boot scanner and Symantec's Bloodhound technology.

¶42 We now turn to a formal analysis of the *res ipsa* inference and computer virus infection as a *res ipsa* case.

III. RES IPSA LOQUITUR: LEGAL DEFINITION AND QUANTITATIVE ANALYSIS

¶43 This section analyzes the legal definition of *res ipsa loquitur* and the inference of negligence under that doctrine. A probabilistic analysis identifies the factors that strengthen an inference of negligence, i.e., create a strong *res ipsa* case. An economic model and an analysis of the technical and economic properties of viruses, as well as virus detection technology, evaluate computer virus infection as a *res ipsa* case.

[42] Nachenberg, *supra* note 1, at 6. *See also* Francisco Fernandez, *Heuristic Engines*, PROCEEDINGS OF THE 11TH INTERNATIONAL VIRUS BULLETIN CONFERENCE, Sept. 2001, at 407-444.

[43] Nachenberg, *supra* note 1, at 7.

[44] Certain byte sequences are, for instance, associated with decryption loops to unscramble a polymorphic virus when an infected routine is executed. If it finds a match, (e.g. the scanner detects the presence of a decryption loop typical of a polymorphic virus), it catalogues this behavior. *Id.* at 7-8.

[45] *Id.* at 9.

A. Legal Definition of Res Ipsa Loquitur

¶44 When deciding the issue of negligence, courts focus on risk-reduction precautions the defendant could have taken, but did not.[46] The plaintiff has the burden to specify an untaken precaution that would have reduced the risk of the accident,[47] and the defendant will be considered negligent if the untaken precaution would have yielded greater benefits in accident reduction than its cost.[48]

¶45 The analytical definition of negligence originated with the Learned Hand formulation in *United States v. Carroll Towing Co.*[49] In *Carroll Towing* , the defendant alleged that the plaintiff was contributorily negligent. The alleged untaken precaution was plaintiff's failure to have an employee onboard a barge, who could have prevented the barge from breaking away and causing an accident. Judge Hand held that the plaintiff was indeed contributorily negligent, based on an assessment of the cost and all foreseeable risks associated with the specific, untaken precaution. The benefit of the reduction in all foreseeable risks that would have resulted from having a bargee on board exceeded the cost of the bargee; hence the barge owner breached his duty of due care.[50] The negligence calculus weighs the cost of an untaken precaution against the value of the reduction in *all* foreseeable risks that the precaution would have achieved, not just the risk that actually materialized.[51]

¶46 Negligence must be proved and will not ordinarily be presumed,[52] unless, in exceptional cases, a statutory provision[53] or contractual relation [54] dictates otherwise. The mere fact that an accident has occurred is not by itself evidence of negligence on the part of anyone. The accident may have been legally "unavoidable," i.e., even reasonable precautions would not have avoided it. What is required is evidence from which a reasonable person may conclude that there is a greater than 50 percent likelihood that the event was caused by negligence.

¶47 Negligence, like any other fact, can be proved by circumstantial evidence. Circumstantial evidence is evidence of a fact from which the fact to be proved may reasonably be inferred. A criminal court may, for instance, base a murder conviction on circumstantial evidence, (e.g., a DNA fingerprint), in the absence of direct evidence (e.g., an eyewitness account). Circumstantial evidence is especially significant in litigation of issues related to information technology, where the

[46] *See, e.g.*, Mark F. Grady, *Untaken Precautions*, 18 J. LEGAL STUD. 139 (1989).

[47] *See, e.g.*, Delisi v. St. Luke's Episcopal-Presbyterian Hosp., Inc., 701 S.W.2d 170 (Mo. 1985) (plaintiff required to specify particular medication he should have been given); Jeffress v. Virginia Ry. & Power Co., 104 S.E. 393 (Va. 1920) (plaintiff identified particular defect in transformer that defendant electric company failed to discover).

[48] *See* Grady, *Untaken Precautions, supra* note 46, at 139-43 (The courts "take the plaintiff's allegations of the untaken precautions of the defendant and asks, in light of the precautions that had been taken, whether some particular precaution promised benefits (in accident reduction) greater than its associated costs.").

[49] United States v. Carroll Towing Co., 159 F.2d 169 (2d Cir. 1947).

[50] Judge Hand summarized the principles of negligence in his decision in *Carroll Towing*: "Since there are occasions when every vessel will break away . . . and . . . become a menace to those about her, the owner's duty . . . to provide against resulting injuries is a function of three variables: (1) The probability that she will break away; (2) the gravity of the resulting injury if she does; (3) the burden of adequate precautions." Denoting the probability by P, the injury L, and the burden B, liability depends on whether B is less than P times L. 159 F.2d at 173.

[51] *See, e.g.*, RESTATEMENT (SECOND) OF TORTS § 281(b), comment e (1965): "Conduct is negligent because it tends to subject the interests of another to an unreasonable risk of harm. Such a risk may be made up of a number of different hazards, which frequently are of a more or less definite character. The actor's negligence lies in subjecting the other to the aggregate of such hazards."; *see also*, In re Polemis & Furness, Withy & Co., (1921) 3 K.B. 560 (C.A.). In Polemis defendant's workman dropped a plank into the hold of a ship, causing a spark which caused explosion of gasoline vapor. The resulting fire destroyed the ship and its cargo. The arbitrators found that the fire was an unforeseeable consequence of the workman's act, but that there was nevertheless a breach of duty. The key to the finding of negligence is the fact that courts base their analysis of untaken precautions on a balancing of all foreseeable risks (not just the risk that materialized) against the cost of the untaken precaution. In finding for the plaintiff, Lord Justice Scrutton stated, "[i]n the present case it was negligent in discharging cargo to knock down the planks of the temporary staging, for they might easily cause some damage either to workmen, or cargo, or the ship [by denting it]." Polemis, at 577.

[52] *See, e.g.,* F. HARPER & F. JAMES, THE LAW OF TORTS § 20.1, at 1108 (1956); In Re Hayden's Estate, 254 P.2d 813 (Kan. 1953); Larkin v. State Farm Mut. Ins. Co., 97 So. 2d 389 (La. 1957); Northwestern Equipment, Inc. v. Cudmore, 312 N.W.2d 347 (N.D. 1981).

[53] Mobile, J. & K.C.R. Co. v. Turnipseed, 219 U.S. 35 (1910).

[54] Osgood v. Los Angeles Traction Co., 137 Cal. 280 (Cal. 1902).

technological complexity and novelty, the frequent absence of an adequate paper trail, and the paucity of legal precedents, make direct proof of negligence difficult.[55]

¶48 A doctrine of circumstantial evidence, known by the Latin name *res ipsa loquitur*, allows an inference of negligence in the absence of clear evidence of specific negligence. The Latin phrase means "the thing speaks for itself," referring to the legal inference that a harmful event may be of such a nature that its *mere occurrence* and the circumstances of the case are sufficient to establish a prima facie case of negligence. It does not require proof of specific negligence, specification of an untaken precaution, or clear evidence of "what went wrong," as long as the elements of *res ipsa loquitur* are satisfied.[56]

¶49 The origins of *res ipsa loquitur* go back to England in the 1860s. One Mr. Byrne was walking down a street in downtown London, minding his own business, when a barrel of flour tumbled out of a warehouse and fell on his head. Byrne was unconscious, and for a while he did not know what had hit him. The next morning he woke up in an infirmary, in a strange bed and in a room that wasn't his, and his head hurt so horribly he probably wished that that was not his either. He contemplated his fate and, in good Anglo-American tradition, decided to file suit against the owner of the warehouse.

¶50 The defendant warehouse owner confidently asserted that it was the plaintiff's obligation to identify an efficient precaution that the workers had failed to take and prove that such untaken precaution would have prevented the barrel from falling. That was the inflexible legal standard at the time, but a major turn of events came when the presiding judge, Baron Pollock, commented, "There are certain cases of which it may be said 'res ipsa loquitur,' and this seems to be one of them." The mere occurrence of the accident is evidence of negligence - specific proof is unnecessary because "the thing speaks for itself." A new doctrine was born.[57]

¶51 Every state recognizes *res ipsa loquitur* in some form, although courts have defined it in different ways. The most commonly articulated conditions for the application of *res ipsa loquitur* are as follows.[58]: It may be inferred that the harm suffered by the plaintiff is caused by the defendant's negligence when (a) the event is of a kind which ordinarily does not occur in the absence of negligence; (b) the defendant had exclusive control over the instrumentality that caused the harm; and (c) the plaintiff did not contribute to her injury or voluntarily assume the risk that led to it.

¶52 Courts, as well as commentators, have argued that the "exclusive control" requirement should be abandoned, or at least interpreted more flexibly. A suggested interpretation that has been widely followed by the courts, is that the plaintiff has to trace the injury to an instrumentality for which the defendant was responsible, even if the defendant was not physically in control of the instrumentality at the time of the accident.[59] The rationale behind this condition is that it is not enough to establish

[55] *See, e.g*, FREED, E VIDENCE, COMPUTERS AND THE LAW 102, 104; BENDER, COMPUTER LAW: EVIDENCE AND PROCEDURE (1982), § 5.01[3].

[56] *See, e.g.,* Fowler v. Seaton, 61 Cal. 2d 681, 687 (Cal. 1964) ("There is no absolute requirement that the plaintiff explain how the accident happened. *Res ipsa loquitur* may apply where the cause of the injury is a mystery, if there is a reasonable and logical inference that defendant was negligent, and that such negligence caused the injury.").

[57] Byrne v. Boadle, 159 Eng. Rep. 299 (Ex. 1863). For a modern case where a keg of beer fell on a pedestrian, see Hake v. George Wiedemann Brewing Co., 262 NE.2d 703 (Ohio 1970).

[58] PROSSER AND KEETON ON THE LAW OF TORTS (West Publ. Co., 5th ed. 1984), § 39, p. 244; This formulation of the necessary conditions for the application of *res ipsa loquitur* was originally stated in Wigmore's treatise on evidence, namely 4 WIGMORE, EVIDENCE (1st ed. 1905) § 2509. This version is frequently cited. *See, e.g.*, Larson v. St. Francis Hotel, 188 P.2d 513, 514 (Cal. 1948); Miles v. St. Regis Paper Co., 467 P.2d 307, 309 (Wash. 1970) (en banc); Ybarra v. Spangard, 154 P.2d 687, 689 (Cal. 1944); Newing v. Cheatham, 124 Cal. Rptr. 193, 199 (Cal. 1975); Raber v. Tumin, 36 Cal. 2d 654, 659 (Cal. 1951); Lynden Transp., Inc. v. Haragan, 623 P.2d 789, 793-94 (Alaska 1981). For statements of the conditions of *res ipsa loquitur* in other states, see Gilbert v. Korvette's, Inc., 327 A.2d 94, 100 (Pa. 1974); Willis v. Terminal R.R. Ass'n., 421 S.W.2d 220, 223 (Mo. 1967); Krebs v. Corrigan, 321 A.2d 558, 559-60 (D.C. 1974).

[59] Prosser, *Res Ipsa Loquitur in California*, 37 CALIF. L. REV. 183, 201 (1949) ("It would be far better, and much confusion would be avoided, if the idea of 'control' were to be discarded altogether, and if we were to say merely that the apparent cause of the accident must be such that the defendant would be responsible for any negligence connected with it."; Giles v. New Haven, 619 A.2d 476 (Conn. 1993), (stating position that "exclusive control of the instrumentality" should be eliminated).

that an accident was negligently caused. It also has to be linked to the defendant by a preponderance of the evidence.[60]

¶53 The evidence need not be conclusive, but should eliminate intervening causes with a greater than 50 percent likelihood. A restaurant employee injured by an exploding beverage bottle, for instance, would have to show that the bottle was not damaged by mishandling after leaving defendant's bottling plant.[61] A virus victim would have to make a plausible case that the infected software was not tampered with by a third party after being released by the defendant and before being acquired by the plaintiff.

¶54 There are cases where the possibility of deliberate tampering by a third party can be ruled out summarily as improbable. Dean Prosser gives the example of a car parked on a steep hill that runs down the hill shortly afterwards. While it is theoretically possible that someone could have tampered with the handbrake, the most likely explanation is the driver's negligence in parking the car.[62] Even though the driver was not physically in control of the car and handbrake at the time of the accident, a court will nevertheless likely find him negligent.

¶55 Under a literal interpretation of condition (c), a negligent plaintiff would be barred from recovery under *res ipsa*. However, almost all jurisdictions today apply comparative negligence, ensuring that plaintiffs will not be barred from recovering under *res ipsa loquitur* because of their own negligence. Juries are typically instructed to compare the relative negligence of the parties and to adjust the damage award accordingly.[63] Therefore, a victim of virus infection will not be disqualified from recovering damages under *res ipsa*, because of lapses, such as failure to make regular backups and to use anti-virus software.

¶56 *Res ipsa loquitur* can only be used where direct proof of negligence is unavailable. The doctrine is therefore most valuable where "the complete story of the plaintiff's injury is unavailable and the factfinder may be tempted to throw up its hands in confusion and to not weigh the evidence before it."[64] It creates an inference of negligence by focusing on key aspects of a plaintiff's case and their respective likelihoods, and analyzing their value as circumstantial evidence in a logical framework. Successful application of *res ipsa* shifts the burden of persuasion to the defendant. The defendant can rebut the inference of negligence by offering exculpatory evidence. If defendant can show, for instance, that the accident could not have been avoided even if all reasonable care had been exercised, then liability may no longer be inferred and the defendant is entitled to a directed verdict.[65] If the defendant fails to rebut, the case is sent to the jury with the instruction that they may, but are not compelled to, infer negligence from the circumstances of the case.[66]

¶57 We now turn to a quantitative model and probabilistic analysis of the *res ipsa* inference of negligence.

B. Quantitative analysis of res ipsa inference

¶58 The most widely accepted formulation of *res ipsa* provides that it may be inferred that the harm suffered by the plaintiff is caused by the defendant's negligence when (a) the event is of a kind which

[60] Marathon Oil Co. v. Sterner, 632 S.W.2d 571 (Tex. 1982); Newing v. Cheatham, 540 P.2d 33, 41 (Cal. 1975) ("The purpose of this requirement is to link the defendant with the probability, already established, that the accident was negligently caused.").

[61] *See, e.g.*, Joffre v. Canada Dry Ginger Ale, Inc., 158 A.2d 631 (Md. 1960); PROSSER *supra*, note 58, at 249.

[62] *Id.*

[63] *See, e.g.*, Emerick v. Raleigh Hills Hospital, 184 Cal. Rptr. 92, 98 (Cal. App. 1982) ("[B]ecause of the advent of comparative fault, *res ipsa loquitur* may apply even if plaintiff's fault contributed to the injury."); Montgomery Elevator Co. v. Gordon, 619 P.2d 66, 70 (Colo. 1980) (contributory negligence does not bar recovery; court should look to comparative fault of the parties); Cyr v. Green Mountain Power Corp., 485 A.2d 1265, 1268 (Vt. 1984); Turk v. H.C. Prange Co., 119 N.W.2d 365, 371-72 (Wis. 1963); John T. Arnold Assocs., Inc. v. Wichita, 615 P.2d 814 (Kan. App. 1980).

[64] Stephen A. Spitz, *From Res Ipsa Loquitur to Diethylstilbestrol: The Unidentifiable Tortfeasor in California*, 65 IND. L. J. 591, 596 (1990).

[65] Oliver v. Union Transfer Co., 71 S.W.2d 478 (Tenn. 1934); Wilson v. Stilwell, 309 N.W.2d 898 (Mich. 1981).

[66] PROSSER AND KEETON ON THE LAW OF TORTS (West Publ. Co., 5th ed. 1984), § 40. *See also*, *Id.* at 258 n.5.

ordinarily does not occur in the absence of negligence; (b) the defendant had exclusive control over the instrumentality that caused the harm; and (c) the plaintiff did not contribute to her injury or voluntarily assume the risk that led to it.

¶59 If we assume that conditions (b) and (c) are satisfied, then the formulation implies that condition (a), the condition, "ordinarily does not occur in the absence of negligence," is sufficient to allow the inference of negligent causation by a preponderance of the evidence. This verbal statement is mathematically ambiguous, i.e., has different plausible mathematical "translations," not all of which imply an inference of negligence by a preponderance of the evidence.[67]

¶60 The phrase, "ordinarily does not occur in the absence of negligence," may be literally interpreted, "given reasonable care, the accident in question should be rare." This is a misleading indicator of the strength of a case as a strong *res ipsa* candidate. An accident may be rare, regardless of the presence or absence of negligence. Hence, its mere occurrence is not necessarily an indicator of negligence. The occurrence of an injury that would be rare in the absence of negligence, but relatively frequent in the presence of negligence, would be a more reliable indicator of negligence.[68]

¶61 Some courts, such as the Oregon Supreme Court in *Brannon v. Wood*,[69] have recognized this concern, while others, unfortunately, have not. In *Brannon* a patient entered a hospital to have a tumor removed from the back of his chest and woke up paralyzed from the waist down, a rare side-effect of emergency treatment of internal hemorrhaging. The Oregon Supreme Court affirmed the trial court's decision to refuse a *res ipsa* jury instruction, stating that "[t]he test is not whether a particular injury rarely occurs, but rather, when it occurs, is it ordinarily the result of negligence."[70] In other cases, expert testimony that an accident would not ordinarily occur if due care were exercised was considered adequate to infer negligence.[71] By this unfortunate reasoning, the fact that a complication is rare, in the presence as well as absence of negligence, would be sufficient to infer negligence.

¶62 We now turn to a probabilistic analysis of the *res ipsa* inference of negligence.

C. Probabilistic model and notation

¶63 This subsection introduces the setting and notation for a probabilistic model of *res ipsa loquitur*. We consider a generic setting in which a software or Internet service provider takes precautions against the transmission of a computer virus, over a time period. The precautions consist of a durable and a non-durable component. A virus scanner with its signature database is a typical durable precaution, while regular maintenance of the database and monitoring of the scanner output are typical complementary non-durable precautions. At the end of the time period, a software product (which may be broadly defined to include a service, such as Internet access) is released, perhaps containing a virus.

¶64 We denote the provider's investment in durable precautions by x, and her investment in non-durable precautions by y. If we represent by r the rate of non-durable precautions, and by z the cost of each repetition, then y = r•z. z may, for instance, represent the cost of one viral signature database update, r the number of updates over a time period, and y the total cost of subscribing to the updating service over the time period.

¶65 We denote the set of all relevant states by a *sample space*,

[67] *See, e.g.*, David Kaye, *Probability Theory Meets Res Ipsa Loquitur*, 77 MICH L. REV. 1456, n.14-18 (1979).

[68] David Kaye, *Probability Theory Meets Res Ipsa Loquitur*, 77 MICH L. REV. 1456, 1461–64 (1979).

[69] Brannon v. Wood, 444 P.2d 558 (Or. 1968).

[70] *See Id.* at 561.

[71] Dacus v. Miller, 479 P.2d 229 (Or. 1971). *See also* Widmeyer v. Southeast Skyways, Inc., 584 P.2d 1, 14 (Alaska 1978); St. Paul Fire & Life Mar. Ins. Co. v. Watkins, 495 P.2d 265, 267 (Or. 1972).

> $\square = \{\square_0, \square_1, \square_2, ..., \square_N\}$. The element $\square_0 \square \square$ represents the state "virus-free," while each of the remaining elements, $\square \backslash \square_0 = \{\square_1, \square_2, ..., \square_N\}$, represents the presence of a particular virus strain. We assume in this model that exactly one element of $\square$ is realized, i.e., the software is either virus-free or infected by exactly one virus strain.

¶66 We define an *event* as the union of several elementary states. The event, "some virus was present," for instance, would be represented by the union of all possible strains, namely V = $\{\square_1, \square_2, ..., \square_N\}$. The interpretation is that, given that event V occurred, all we know is that *some* virus strain was present, i.e., some element of V.

¶67 The set, E = $\{\square_1, \square_3, \square_7\}$, denotes the event "occurrence of $\square_1$ *or* $\square_3$ *or* $\square_7$"

¶68 To facilitate a formal probabilistic analysis of events, we define the *sigma-algebra* on $\square$, denoted $\square$, as the set of all subsets of $\square$. Each member of $\square$ is an event. The previously defined subset of $\square$, E = $\{\square_1, \square_3, \square_7\}$ is a typical member of $\square$.

¶69 The transmission of certain virus strains will be avoided, depending on the precautions taken. This is formalized as follows. V(x, y)$\square \square$ denotes the set of virus strains avoided with durable precautions at level x, and non-durable precautions at y. If, for instance, V(x, y) = $\{\square_1, \square_3, \square_7\}$, then we say that durable precautions at level x, combined with non-durable precautions at level y, will prevent infection by virus strains $\square_1$, $\square_3$ and $\square_7$. An updated version of Norton Antivirus, for instance, is capable of detecting virtually all currently known virus strains, including Melissa and ILoveYou.

¶70 Probabilities are assigned to events in $\square$ by a probability measure, denoted $\square$. We denote, for instance, the probability of the event $\{\square_1, \square_3, \square_7\}$ by $\square[\{\square_1, \square_3, \square_7\}]$. This quantifies the probability that any one of the states, $\square_1$, $\square_3$, or $\square_7$ will be realized. The expression $\square$[V(x, y)] + $\square[\square_0]$ can therefore be interpreted as the probability that the provider's product will be virus-free, given durable and non-durable anti-viral precautions (x, y).

¶71 We assume that y ≤ y' implies that V(x, y) $\square$ V(x, y'), for all x.[72] The plausibility of this assumption can be motivated as follows. Suppose x represents an investment in a virus scanner; y represents the cost of a commitment to update the signature database weekly, and y' represents the cost of a commitment to update the database more frequently, say, twice weekly. The set of viruses preventable with the more intense precautions (x, y'), then, is at least as great as the set preventable with (x, y), i.e., V(x, y') contains V(x, y).

¶72 The combination of the sample space ($\square$), the sigma-algebra ($\square$), and the probability measure ($\square$), denoted [$\square$, $\square$, $\square$], is defined as the *probability space* of the mathematical model.

D. Probabilistic analysis of the res ipsa inference

¶73 *Res ipsa loquitur* allows the inference, by a preponderance of the evidence, that an accident was negligently caused, conditional upon the fact of its occurrence and the circumstances of the case. In mathematical notation, Prob(N/I) > 0.5, where "Prob" denotes probability, "N" denotes the event "accident negligently caused," and "I" denotes the fact of the accident's occurrence and its circumstances.

¶74 We denote the set of strains that would be prevented by due care precaution level, (x*, y*), by V(x*, y*). A negligently transmitted virus strain is by definition one that (i) *would* have been prevented, *had* due care been taken, i.e., an element of V(x*, y*), *and* (ii) was *not* prevented because actual care (y') fell below the due care level (y*), i.e., the strain is an element of the complement of V(x*, y'), where y' < y*.[73] Hence, the event "negligent transmission of virus strain" corresponds to

[72] This is read as "V(x, y) is contained in V(x, y') for all x," i.e., V(x, y') contains at least the elements of V(x, y).

[73] It is shown, *infra*, that a rational, profit-maximizing provider will invest in durable precautions at the due care level, x*, but may fall short of the due care level of non-durable precautions, i.e. the actual y may be less than y*.

the realization of a state in the intersection of the sets V(x*, y*) and the complement of V(x*, y'). In mathematical notation: $V(x^*, y^*)$ □ $V(x^*, y')^c$. This notation can be simplified by applying the De Morgan rules of Boolean logic[74]:

$V(x^*, y^*)$ □ $V(x^*, y')^c = [V(x^*, y^*)^c$ □ $V(x^*, y')]^c$;

commonly denoted $V(x^*, y^*)$ □ $V(x^*, y')$.[75]

¶75 The *a priori* probability of negligent transmission of a virus strain, denoted Prob(N), is calculated by applying the probability measure, □, to the set

$V(x^*, y^*)$ □ $V(x^*, y')^c$, namely:

Prob(N) = □$[V(x^*, y^*)$ □ $V(x^*, y')^c]$

= □$\big[[V(x^*, y^*)^c$ □ $V(x^*, y')]^c\big]$

= 1 - □$[V(x^*, y^*)^c$ □ $V(x^*, y')]$

= 1 - □$[V(x^*, y^*)^c]$ - □$[V(x^*, y')]$, since the sets $V(x^*, y^*)^c$ and $V(x^*, y')$ are disjoint;[76]

= □$[V(x^*, y^*)]$ - □$[V(x^*, y')]$.

¶76 A transmitted virus can be either of the negligently or unavoidably transmitted variety. Negligent transmission, i.e transmission of a virus that would have been avoided had due care been exercised, corresponds to the set $V(x^*, y^*)$ □ $V(x^*, y')$, as derived above. A virus strain is "unavoidable" if even due care is inadequate to eliminate it. We define the set of unavoidable virus strains by $V_U = V(x^*, y^*)^c \backslash$ □$_0$. In other words, unavoidable strains fall outside the set V(x*, y*), that would have been avoided by due care, and trivially excludes the "virus-free" state, □$_0$.

¶77 The "avoidable to unavoidable error ratio" is defined as the ratio of the *a priori* probabilities of infection by an avoidable and an unavoidable strain, respectively, namely $\frac{□[V(x^*,□y^*)]}{□□[V_U]}$.

¶78 The event "virus transmitted" can therefore be represented as the union of the sets "negligently transmitted" and "unavoidably transmitted," namely:

$[V(x^*, y^*)$ □ $V(x^*, y')]$ □ V_U.

To keep the notation manageable, we define:

□$_1 := V(x^*, y^*)$ □ $V(x^*, y')$; representing the event, "virus negligently transmitted;"

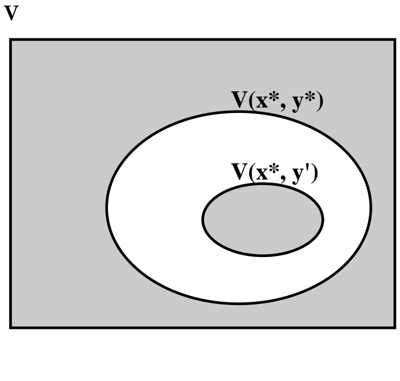
□$_2 := [V(x^*, y^*)$ □ $V(x^*, y')]$ □ V_U; representing the event, "virus transmitted."

¶79 The probability that a virus was transmitted through the negligence of the software provider, given the fact of the transmission, denoted Prob(N/I), can therefore be written:

[74] *See, e.g.*, ERWIN KREYSZIC, INTRODUCTORY FUNCTIONAL ANALYSIS WITH APPLICATIONS, app. I, at 612 (1989).

[75] *See, e.g.*, H.L. ROYDEN, REAL ANALYSIS 13 (3d ed. 1988).

[76] Since $y' < y^*$, $V(x^*, y')$ □ $V(x^*, y^*)$; hence, $V(x^*, y')$ and $V(x^*, y^*)^c$ are disjoint, represented, respectively, by the inner and outer colored areas.



$$\text{Prob}\{\square\ \square\ \square_1/\square\ \square\ \square_2\} = \frac{\text{Prob}\{\square_1 \square\ \square_2\}}{\text{Prob}\{\square_2\}} = \frac{\text{Prob}\{\square_1\}}{\text{Prob}\{\square_2\}}$$

$$= \frac{\square[V(x^*, y^*)] - \square[V(x^*, y')]}{\square[V(x^*, y^*)] - \square[V(x^*, y')] + \square[V_U]}$$

$$= \frac{1}{1 + \dfrac{\square[V_U]}{\square[V(x^*, y^*)] - \square[V(x^*, y')]}}$$

$$= \frac{1}{1 + \dfrac{\square[V_U]}{\square[V(x^*, y^*)]} \Big/ \left[1 - \dfrac{\square[V(x^*, y')]}{\square[V(x^*, y^*)]}\right]}.$$

¶80 This expression is increasing in both $\dfrac{\square[V(x^*, y^*)]}{\square[V_U]}$ as well as $\left[1 - \dfrac{\square[V(x^*, y')]}{\square[V(x^*, y^*)]}\right]$.

¶81 The first term, $\dfrac{\square[V(x^*, y^*)]}{\square[V_U]}$, has been defined as the avoidable to unavoidable error ratio. The second term, $\left[1 - \dfrac{\square[V(x^*, y')]}{\square[V(x^*, y^*)]}\right]$, is a measure of the ex ante probability that a virus will be negligently transmitted.

¶82 An intuitive interpretation of the second term can be motivated as follows. We have shown that the (ex ante) probability that a virus will be negligently transmitted equals Prob(N) = $\square$[V(x*, y*)] - $\square$[V(x*, y')]. The courts fix due care at (x*, y*). If the defendant performs at this level of care, liability will be avoided, as the probability $\square$[V(x*, y*)] - $\square$[V(x*, y*)] = 0.[77] The farther defendant's non-durable precaution level falls below y*, the smaller $\square$[V(x*, y')], and the larger Prob(N) becomes.

¶83 Given that (x*, y*) and $\square$[V(x*, y*)] are fixed, this is equivalent to saying that Prob(N) increases with $\left[1 - \dfrac{\square[V(x^*, y')]}{\square[V(x^*, y^*)]}\right]$.

¶84 According to the foregoing analysis, the following factors strengthen the *res ipsa* inference:

1. A high avoidable-to-unavoidable error ratio.

¶85 An avoidable error is one that is preventable through due care. An unavoidable error is one that even due care would not have prevented. The transmission of a virus in V(x*, y*), for instance, constitutes an avoidable error, i.e. a virus strain that a reasonably careful provider would detect and eliminate. An example of an unavoidable virus is an unknown, complex strain that could only be detected and eliminated at unreasonably high cost, e.g. through expensive and sophisticated scanning techniques based on artificial intelligence technology.

¶86 This result sheds light on an appropriate interpretation of the *res ipsa* condition: "does not ordinarily occur in the absence of negligence."[78] This condition literally says that the probability of an accident, given due care, should be "small," i.e., Prob{I/R} is "small," where the symbol "I" denotes the occurrence of an accident, and "R" denotes "reasonable care." We have argued that a

[77] Assuming that the culprit virus can be identified, the defendant will be exonerated if the identified strain turns out to be outside of V(x*, y*).

[78] *See supra* note 58 and accompanying text.

small value for Prob{I/R} *in the absolute sense* is insufficient to justify a general inference of negligent causation.[79] An event that is seldom caused by negligence, and is equally rare in the absence or presence of negligence, will satisfy this requirement in the absolute sense. An event that is relatively rare in the absence of negligence and significantly more likely in the presence of negligence is a more likely candidate for a *res ipsa* inference. Its occurrence is correlated with negligence--hence, a superior indicator of negligence. Such an accident will exhibit a relatively high ratio $\frac{\text{Prob}\{I/\text{zero precautions}\}}{\text{Prob}\{I/R\}}$ We now show that this ratio is related to and, in fact, increases with the avoidable to unavoidable error ratio.

¶87 Assuming that zero precautions avoids zero viruses,

¶88
$$\frac{\text{Prob}\{I/\text{zero precautions}\}}{\text{Prob}\{I/R\}} = \frac{\square[V(x^*, y^*)] + \square[V_U]}{\square[V_U]}$$
$$= \frac{\square[V(x^*, y^*)]}{\square[V_U]} + 1.$$

¶89 The left hand side quantifies the relative likelihood of an accident in the absence of precautions compared to the likelihood in the presence of due care. The right hand side is the avoidable to unavoidable ratio, plus a constant.

¶90 This equation shows mathematically that a high avoidable to unavoidable error ratio implies that the accident (e.g. presence of a virus) would be rare in the presence of reasonable care, but frequent in the presence of negligence. The latter is an appropriate interpretation of the *res ipsa* condition, Prob{I/R} is "small," namely Prob{I/R} = $\square[V_U]$ must be small *in relation to* $\square[V(x^*, y^*)]$.

¶91 The result that a high avoidable-to-unavoidable error ratio contributes to a strong *res ipsa* case also gives us a theoretically sound and intuitively plausible interpretation of the crucial *res ipsa* condition: "does not ordinarily occur in the absence of negligence."

2. A large a priori probability of negligent viral transmission.

¶92 The greater the defendant's *a priori* propensity for negligence (i.e. prior to knowledge of viral infection), the stronger the inference of negligence when the fact of the viral infection is added to the circumstances of the case.

¶93 The next section presents an economic analysis of virus prevention, and shows that a rational, profit-maximizing software provider has an economic incentive to take precautions at a level that maximizes expected profitability, but that falls below the legal standard of due care. This creates a high *a priori* probability of negligence, contributing to a strong *res ipsa* case.

IV. ECONOMIC ANALYSIS OF VIRUS PREVENTION

¶94 The second major factor that creates a strong *res ipsa* case is a high *a priori* probability of negligent viral transmission. This prompts the question, why would a rational profit-maximizing software or service provider engage in negligent behavior? This section presents an economic analysis showing that providers have an economic incentive to take anti-viral precautions at a level below the legally required level of due care. A significant discrepancy between actual and due care precaution levels creates a high *a priori* probability of negligent viral transmission. The concept of efficient negligence, pioneered by Dean Mark Grady, is formalized here in a virus context.[80]

[79] *See supra* Section III, subsection B.

[80] Grady, *supra* note 11.

A. Why is Prob(N) > 0? Compliance error and negligence.

¶95 A high *a priori* probability of negligence contributes to a strong *res ipsa* case. Results in the law and economics literature, however, have established that there will be no negligent behavior under a negligence rule of liability, in the absence of errors about legal standards, when precaution is not random, and when private parties have identical precaution costs.[81] It seems therefore, that the frequent occurrence of negligence in society must be explained in terms of non-uniform precaution costs, errors by courts and private parties about the relevant legal standards, or that precaution has a random or stochastic component.

¶96 Dean Grady has argued that none of these theories explain the prevalence of negligence entirely satisfactorily. Grady proposes a theory according to which there is a pocket of strict liability within the negligence rule.[82] A rational injurer may find an occasional precautionary lapse preferable to perfect compliance with the legal standard of due care. The frequency of such lapses will increase as the due care standard becomes more burdensome. The occasional lapse is rational and profit maximizing, but will nevertheless be classified as negligence by the courts because of the courts' inability to distinguish between efficient and inefficient lapses.[83] The term "pocket of strict liability" therefore refers to "efficient negligence," i.e., behavior that is efficient from the point of view of the injurer, yet negligent in the view of the court. We argue that this "pocket of strict liability" theory is particularly applicable to an analysis of negligence in anti-viral precautions.

¶97 The level of investment in durable and non-durable anti-virus precautions required by negligence law is determined according to the Learned Hand formula. Scanners, for instance, come in a variety of degrees of sophistication (and cost), ranging from basic systems that detect only known strains, to heuristic artificial intelligence-based systems capable of detecting polymorphic viruses and even unknown strains. The optimal Learned Hand level of investment in scanning technology would be determined by balancing the cost of acquiring and operating the technology against the expected harm avoided. The optimal frequency of viral database updating is determined similarly.

¶98 The courts require perfectly consistent compliance with the Learned Hand precautions to avoid a finding of negligence. If, for instance, the courts require a database to be updated twice daily, then even one deviation, such as a skipped update, would be considered negligent.[84] When the courts apply the Hand formula to determine an efficient precaution level and rate, the calculation weighs the costs and benefits of the precaution *each time* it is performed but ignores the cost of consistently performing it *over time*. Consider a numerical example. Suppose the cost of a daily update is $10, and the marginal benefit of the update is $11. Failure to perform even one such update would be viewed as negligence by the courts. Over, say, 300 days, the courts expect 300 updates, because each of those updates, by itself, is Learned Hand efficient. However, the courts do not consider the cost of consistency, i.e. of *never* forgetting or lapsing inadvertently. Human nature is such that over a 300-day period, the person in charge of updating will occasionally inadvertently fail to implement an update.

¶99 Human nature, being what it is, dictates that perfection is (perhaps infinitely) expensive.[85] Perfect consistency, i.e. ensuring that 300 updates will actually be achieved over 300 days, would

[81] *See,* Grady *supra* note 15, at 889-91; *see also supra* references cited in note 5.

[82] *See Id.* at 897.

[83] *See infra* next subsection, "Why do courts insist on perfect compliance?".

[84] In *Kehoe v. Central Park Amusement Co.*, 52 F.2d 916 (3d Cir. 1931), an amusement park employee had to apply a brake to control the speed of the car each time the roller coaster came around. When he failed to do so once, the car left the track. The court held that the compliance error by itself constituted negligence, i.e. the court required perfect compliance and considered anything less as negligence. 52 F.2d 917 ("If the brake was not applied to check the speed as the car approached...it was clear negligence itself.") For other cases, *see* Grady, *supra* note 15, at 900-902. In *Mackey v. Allen*, 396 S.W.2d 55 (Ky. 1965) plaintiff opened a "wrong" exterior door of a building and fell into a dark storage basement. The court held the two tenants of the building liable for failing to lock the door. *But cf* Myers v. Beem, 712 P.2d 1092 (Colo. Ct. App. 1985) (holding, in an action brought against an attorney for legal malpractice that lawyers are not required to be infallible.).

[85] *See, e.g.*, IVARS PETERSON, FATAL DEFECT 194 (1995) ("Even under the best of circumstances, our brains don't function perfectly. We do forget. We can be fooled. We make mistakes. Although complete failures rarely occur, neural systems often

require additional measures, such as installing a monitoring device alerting the operator to a lapse, or perhaps additional human supervision, all of which are costly. Even assuming (unreasonably) that such measures would assure consistency, their cost may nevertheless be prohibitive to a rational software provider. Suppose, for instance, that such a measure would add an additional $2 to the cost of an update. The marginal cost of an update ($12) is now more than the marginal benefit ($11). Hence, perfect consistency is not in society's interest. An occasional lapse is also reasonable from the viewpoint of the software provider: the marginal cost of perfect consistency is greater than the marginal increase in liability exposure due to efficient negligence.

¶100 Efficient lapses can be expected to become more likely and more frequent, the more demanding and difficult the Learned Hand non-durable precaution rate, i.e. the more expensive perfect consistency becomes. Any lapse in compliance would, however, be considered negligence by the courts, because their negligence calculus does not take the cost of consistency into account. Following Dean Grady, we define an efficient deviation from perfect compliance with the (Learned Hand) non-durable precaution rate as a "compliance error."[86]

¶101 Since failure to invest in durable precautions, at least at the Learned Hand level, and compliance errors both count as negligence in the view of the courts, most negligent behavior on the part of rational, profit-maximizing software and service providers will be the result of compliance errors. Investing in durable precautions up to the efficient Learned Hand level is profit-maximizing because such investment reduces the provider's liability exposure by more than it costs. A compliance error is efficient due to the high cost of perfect consistency, and thus likewise profit-maximizing. Negligent behavior will therefore be more likely the more likely compliance errors are. The more burdensome the rate of compliance and the more difficult and expensive perfect compliance with the non-durable precaution rates, the greater the expected number of compliance errors,[87] and, hence, the greater the *a priori* probability of negligent causation of an accident, and the resulting strength of the *res ipsa* inference.

¶102 We now turn to the question, given that perfect compliance is inefficient, why do courts insist on it?

B. Why do courts insist on perfect compliance?

¶103 A major reason for the courts' insistence on perfect compliance, in spite of the inefficiency of such perfection, is that it is either impossible or too expensive to determine whether any given deviation from perfect compliance is efficient. Who can judge, for instance, whether a software provider or web site operator's mistake or momentary inattentiveness was an economic or uneconomic lapse? Courts therefore do not acknowledge efficient non-compliance where it is difficult to distinguish between efficient and inefficient non-compliance.

¶104 The policy rationale behind the courts' insistence on perfect compliance was expressed by Lord Denning in *Froom v. Butcher*:[88] "The case for wearing seat belts is so strong that I do not think the law can admit forgetfulness as an excuse. If it were, everyone would say 'Oh, I forgot.'"[89] Instead of incurring the considerable measurement cost to distinguish between efficient and inefficient failures to comply, courts simply equate any and all non-compliance to negligence.[90]

¶105 Courts tend to be forgiving, however, where the cost of ascertaining the efficiency of non-compliance is low or zero. In cases where the deviation is demonstrably efficient or unavoidable,

suffer local faults.").

[86] Grady, *supra* note 15.

[87] Grady, *supra* note 15, at 922.

[88] 3 All E.R. 520, 527 (C.A.) (1975).

[89] *Id.*

[90] *See also* Grady, *supra* note 15, at 906; W. LANDES AND R. POSNER, THE ECONOMIC STRUCTURE OF TORT LAW 73 (1987).

such as an accident resulting from a defendant's (provable) temporary physical incapacitation, courts have not imposed liability.[91]

¶106 We now turn to a formalization of these arguments, namely an economic analysis of compliance with non-durable anti-viral precautions.

C. An economic model of compliance error

¶107 This subsection presents a formal economic model of compliance error in the context of virus prevention. The analysis shows that a rational software or service provider will invest in durable anti-virus precautions at the due care level required by negligence law. However, the provider will invest in non-durable precautions at a level below the due care level. It is cheaper for the provider to spend less on non-durable precautions and to incur the risk of liability costs, rather than incurring the even higher cost of achieving perfectly consistent compliance with the legally-imposed due care standard.

¶108 In the notation introduced in Section III, investment in durable and non-durable antiviral precautions, (x, y), avoids the set of viral strains denoted V(x, y). This reduces the *a priori* probability of viral infection to $[1 - \square(\square_0) - \square[V(x, y]]$. The term $\square[V(x, y)]$ represents the contribution of inputs (x, y) to the reduction of virus infection risk. We assume the "production function" relating inputs, (x, y), to the "output," $\square[V(x, y)]$, can be represented by the relation: $\square[V(x, y)] = Ax^{\square} + By^{\square}$. We denote by D the expected damages from the presence of a virus in a system.[92]

¶109 The positive economic theory of breach of duty posits that negligence law aims to minimize social cost.[93] Under this theory, a software provider would escape liability by taking the cost-minimizing amount of precaution. For the purpose of this article, we define social cost as the sum of precaution cost and expected harm:

$$\begin{aligned}\text{Total social cost} &= \text{Total investment in precautions + Expected harm}\\ &= x + y + [1 - \square(\square_0) - Ax^{\square} - By^{\square}]\bullet D\end{aligned}$$

The due care standard that minimizes this expression is:

$$x^* = (\square AD)^{1/(1\square\square)},\ y^* = (\square BD)^{1/(1\square\square)}.$$

¶110 We denote the rate of efficient non-compliance by □ In other words, an investment, y, in non-durable precautions is in practice effectively reduced to y(1-□) due to efficient lapses. Suppose, for instance, the due care standard required a software provider to update a virus signature database twice daily, or sixty times monthly. Subscribing to a service that provides new signatures at the rate of sixty per month, costs an amount y. However, because of efficient lapses (the provider may inadvertently fail, perhaps forget, to implement some of the updates), only amount y(1-□) effectively contributes to reducing the probability of viral infection.

¶111 If, for instance, □ = 10 percent, the provider will lapse 6 times, and implement 54 updates per month. With a lower required compliance rate, fewer lapses will occur. With, for instance, a due care compliance rate of 6 updates per month, the software provider will achieve an average of 5.4 (i.e., lapse only six times in ten months).

¶112 We now turn to a model of the software or service provider's incentives to invest in durable and non-durable precautions. A realization, □ □ □, is drawn. If the provider's software transmitted a

[91] *See, e.g.*, Grady, *supra* note 15, at 897 n.26. *See also* Ballew v. Aiello, 422 S.W.2d 396 (Mo. Ct. App. 1967) (finding defendant not liable for negligence because he was half asleep at the time he was allegedly negligent.); Grady, *supra* note 15, at 905 n.59. ("For faints and other unusual slips, it is possible for courts to judge whether they should have been avoided. Indeed, courts' measurement of unusual slips reintroduces the negligence component back into the negligence rule.").

[92] Section VI, *infra*, presents an analysis of the nature and scope of damages from virus infection.

[93] John Brown, *Toward an Economic Theory of Liability*, 2 J. LEGAL STUD. 323 (1973); W. Landes and R. Posner, *A Theory of Negligence*, 1 J. LEGAL STUD. 29 (1972); S. Shavell, *Strict Liability Versus Negligence*, 9 J. LEGAL STUD. 1 (1980).

virus, i.e. $\square \neq \square_0$, defendant's negligence depends, in principle, on whether the transmitted virus strain belongs to V(x*, y*). If $\square\ \square$ V(x*, y*), i.e. the software had been infected by an avoidable strain, then the court should impose liability, and award damages. If $\square\ \square$ V(x*, y*), the defendant should be acquitted.

¶113 We assume that a defendant will escape liability if it can be verified that $\square\ \square\ V_U$ and conversely, that he will be held liable if it can be verified that $\square\ \square$ V(x*, y*). In addition, if the identity of the transmitted virus cannot be verified, the defendant may be held liable by, for instance, invoking *res ipsa loquitur*. We denote the probability of this event by P_0.

¶114 Ignoring second-order terms, the total expected cost faced by the provider, including liability costs, can be written: $x + y + [\square[V(x^*, y^*) - \square[V(x, y(1-\square))] + P_0]\bullet D$.

¶115 Substituting $\square[V(x, y(1-\square))] = Ax^{\square} + B(1-\square)^{\square}y^{\square}$, and minimizing the total cost expression, gives the due care solution for x, namely $x^* = (\square AD)^{1/(1\square\square)}$, as before, but a below-due care solution for y, namely y' < y*. The software provider who perceives a higher cost of compliance with non-durable precautions than the courts acknowledge consumes less non-durable precaution, i.e. commits an efficient compliance error. Solving the minimization problem for y:

$$y' = (\square BD)^{1/(1\square\square)}\bullet(1-\square)^{\square/(1\square\square)}$$
$$= (1-\square)^{\square/(1\square\square)}\bullet y^*$$
$$< y^*.$$

¶116 This result suggests that a software provider will find it optimal to deviate from the court-imposed non-durable precaution level, y*. It is cheaper for the provider to spend less on non-durable precautions and to incur the risk of liability costs, rather than incurring the even higher cost of achieving perfectly consistent compliance with the legally-imposed due care standard, y*.

¶117 We have shown[94] that the probability of negligent virus transmission, given the fact of the transmission is

$$\text{Prob(N/I)} = \frac{1}{\square 1\square+\square\dfrac{\square[V_U]}{\square[V(x^*,\square y^*)]\square\square\square[V(x^*,\square y')]}}.$$

¶118 Substituting the expressions for x* and y':

$$= \frac{1}{\square 1\square+\square\dfrac{1\square-\square\square[\square_0]\square-\square A(\square AD)^{\square/(1-\square)}\square-\square B(\square BD)^{\square/(1-\square)}}{\square B(\square BD)^{\square/(1-\square)}[1\square-\square(1\square-\square\square)^{\square/(1-\square)}]}}.$$

¶119 This expression shows that the strength of an inference of negligence increases in D, A, □, B, and □.

¶120 D has been defined as the danger level (i.e. expected damage), associated with virus infection. The constants A, B, □ and □ can be interpreted as indices of sophistication and productivity of anti-viral technology. The higher the numerical values of A and □, for instance, the greater the reduction in infection risk achievable by a given investment in durable precautions. Writing the decrease in probability of infection achieved by a marginal increase in x (marginal productivity of the precaution) as $\square\bullet Ax^{\square-1}$, x > 1, illustrates the interpretation of □ and A as indices of the productivity of durable precaution technology. Constants B and □, likewise, represent productivity and technological sophistication of non-durable anti-viral precautions. Hence, a high danger level, coupled with sophisticated and productive virus detection technology, contribute to a strong *res ipsa* inference of negligence.

[94] *See supra* Section III.

¶121 Prior results have shown that (i) a high “avoidable to unavoidable error ratio”, and (ii) a high *a priori* probability of negligent transmission, create a strong *res ipsa* inference of negligence. This section has shown that high values for D, A, □, B, and □, likewise, create a strong *res ipsa* inference. We now show that these sets of results are consistent, namely that both (i) and (ii) are increasing in D, A, □, B, and □.

¶122 High danger rates require intense precautions and high precaution rates, which leave little room for unavoidable error. The result is a high “avoidable to unavoidable error ratio.” In mathematical terms, a high D gives a high (x*, y*), a high □[V(x*, y*)], and low □[V_U]. The result is a high avoidable to unavoidable error ratio, $\frac{\square[V(x^*,\square y^*)]}{\square\square[V_U]}$. The dependence of this ratio on D, A, □, B, and □ can be illustrated mathematically, as follows:

¶123
$$\frac{\square[V(x^*,\square y^*)]}{\square\square[V_U]} = \frac{\square[V(x^*,\square y^*)]}{1\square-\square\square(\square_0)\square-\square\square[V(x^*,\square y^*)]}$$

$$= \frac{1}{\frac{1\square-\square\square(\square_0)}{\square[V(x^*,\square y^*)]}\square-\square 1}.$$

¶124 Writing □[V(x*, y*)] = $A(\square AD)^{\square/(1\square\square)} + B(\square BD)^{\square/(1\square\square)}$, illustrates that the “avoidable to unavoidable error ratio” increases in A, □, B, □, and D.

¶125 High values of B, □, and D also create a high *a priori* probability of negligent viral transmission, the second major factor that creates a strong *res ipsa* case. Intense non-durable precaution rates and levels, (i.e. a high y*), give a high compliance error rate, because of the difficulty of achieving it consistently. This statement can be verified mathematically:

$$\text{Probability of negligent viral transmission} = \square[V(x^*, y^*)] - \square[V(x^*, y')]$$
$$= B(\square BD)^{\square/(1\square\square)}[1 - (1 - \square)^{\square/(1\square\square)}] = B(y^*)^{\square}[1 - (1 - \square)^{\square/(1\square\square)}],$$

which is increasing in B, □, D, as well as y*.

¶126 Having identified the factors that make virus infection a strong *res ipsa* case, we now turn to the question: to what extent are these factors actually present in the real world of viruses?

V. VIRAL INFECTION AND CIVIL LIABILITY

¶127 The analysis in Sections III and IV has shown that (i) sophisticated and productive virus defense technology, (ii) durable precautions complemented by high rates and levels of non-durable precautions, and (iii) a high danger level, all contribute to a high avoidable to unavoidable error ratio and a high *a priori* probability of negligence. These factors, in turn, create a strong *res ipsa* case. This section analyzes these factors in a computer virus context.

¶128 A generic computer malfunction does not present a strong *res ipsa* case because the avoidable to unavoidable error ratio remains low. Computer virus transmission, in contrast, exhibits a comparatively high avoidable to unavoidable error ratio. Sophisticated state-of-the-art virus detection and elimination technology makes the detection of a large proportion of virus strains technologically feasible. The Learned Hand formula, applied to anti-virus precautions, suggests that the high danger level of virus infection, the high (and rapidly increasing) prevalence of viruses, and the modest cost of anti-virus precautions, create a legal duty to implement sophisticated and effective anti-virus technology that is capable of avoiding a large proportion of all virus strains. The result is a high avoidable to unavoidable error ratio.

¶129 Other aspects of virus and virus detection technology contribute to the strength of viral infection as a *res ipsa* case. Unique aspects of viral code, such as the presence of a digital signature, enables virus detection software to reliably identify a virus as the culprit when it is responsible for a computer malfunction. This transforms the virus victim's case from one with a low avoidable to unavoidable error ratio (general malfunction) to one with a high avoidable to unavoidable ratio (malfunction due to virus). Finally, a high expected compliance error rate creates a high *a priori* probability of negligent viral transmission.

¶130 The remainder of this section develops and formalizes these arguments.

A. Generic computer malfunction as res ipsa case

¶131 It is often difficult or impossible to determine whether a computer program error is due to the programmer's lack of due care, or to factors beyond her control. This uncertainty is the result of the complex interaction between hardware, systems software, application software, and the human factor.[95]

¶132 System software manages the computer system and performs housekeeping functions, including facilitating the creation of application programs.[96] Because of the close synergy between system software and application software, it is often difficult to distinguish whether a program error originated in the system or application software. A system program error may interfere with the operation of an application program, giving the appearance of a bug in the latter. Furthermore, system software is quite complex and written in symbolic language, making identification of the cause of an error daunting to a plaintiff who is usually less familiar with the system than the defendant programmer or software provider.[97]

¶133 The cause of an error in an application program usually cannot be inferred with any degree of reliability from a similar error in another system. Because application software is tailor-made to the needs of an individual system, generally any two programs that perform a similar function on two different systems will not be alike.[98]

¶134 The hardware of a computer system includes all physical components of the system, such as input, output and storage devices, and the central processing unit (CPU). The CPU is responsible for data processing and computation.[99] Hardware may spontaneously malfunction because of numerous factors including temperature variations, humidity changes, static electricity, electronic noise, electromagnetic radiation, and power fluctuations.[100] Such transient failures, or soft errors, are usually beyond the control of the software provider and, by their ephemeral nature, very difficult to detect.[101] Transient failures often make it difficult to distinguish between hardware and software failure as the origin of a computer error. Power fluctuations or static electricity may alter data stored in memory, so that a hardware error may be mistaken for a software error. Hardware constraints, such as rounding errors that result in computational anomalies, may also masquerade as software defects.[102] Data validation routines designed to ensure that input data conform to the right format,

[95] Daniel J. Hanson, *Easing Plaintiffs' Burden of Proving Negligence for Computer Malfunction*, 69 IOWA LAW REVIEW 241, 243-48 (1983).

[96] D. BENDER, COMPUTER LAW: EVIDENCE AND PROCEDURE § 2.06[2], at 2-117 (1982).

[97] CHO, *supra* note 8, at 240 (discussing the complexity of system software).

[98] *See* D. BENDER, COMPUTER LAW: EVIDENCE AND PROCEDURE § 5.01[3][a], at 5-12 (1982).

[99] HRUSKA, *supra* note 28, at 113; D. BENDER, COMPUTER LAW: EVIDENCE AND PROCEDURE §2.02, at 2-7 to 2-9 (1982).

[100] D. BENDER, COMPUTER LAW: EVIDENCE AND PROCEDURE § 2.05(6), at 2-104, § 8.02(2)(a)(i)(B), at 8-22 to 8-25 (1982); CHO, *supra* note 8, at 239; BORIS BEIZER, SYSTEM SOFTWARE TESTING AND QUALITY ASSURANCE 29 (1984) ("One of the unfortunate side effects of large-scale integrated circuitry stems form the use of microscopic logic elements that work at very low energy levels. Modern circuitry is vulnerable to electronic noise . . . stray alpha particles and other noxious disturbances. [A]lpha particle hits that can change the value of a bit are a serious problem . . . .").

[101] Beizer, *supra* note 100, at 29-30. *See also* JOHN ZARRELLA, SYSTEM ARCHITECTURE 216 (1980).

[102] A mishap in European Space Agency's Ariane 5 was discovered to be due to precision error in an attitude sensor. Documented at http://catless.ncl.ac.uk/Risks/18.29.html#subj16.

may nevertheless be frustrated by interfering events that have nothing to do with the software. Data can be corrupted, for instance, when electronic noise changes the value of a bit.

¶135 Software may fail because of human errors, such as faulty data input, which may not always be ex-post verifiable. It may not be possible, for instance, to exactly reproduce the format and quality of data input into the system in order to verify that faulty data (and not a programming bug) caused the error.

¶136 The difficulty in attributing an unexplained computer error to a breach of duty by the hardware manufacturer, system programmer, application software provider, or human operator makes the error a weak *res ipsa* case against any of these parties. The probability that such an error is due to, say, programmer negligence, is low because of the many alternative explanations for the error that have nothing to do with programmer negligence.

B. Technology and the avoidable to unavoidable error ratio

¶137 The avoidable to unavoidable error ratio of an accident measures the proportion of potential causes of the accident that would be avoidable through due care. This ratio would be high if the elimination of most causes were not only technologically feasible, but also cost-effective, placing their elimination within the scope of due care.

¶138 In this subsection we argue that state-of-the-art antivirus technology is capable of detecting virtually 100 percent of known virus strains and in excess of 80 percent of unknown strains. In the next subsection we argue that the high danger level associated with virus infection and relatively low cost of anti-virus software, make efficient precautions cost-effective, resulting in a high avoidable to unavoidable error ratio. A generic computer malfunction, in contrast, has a low avoidable to unavoidable error ratio. Even heroic precautions on the part of the programmer cannot eliminate many, perhaps most, errors that cause such malfunctions.

¶139 Antivirus software became available soon after the first appearance of computer viruses, and has become increasingly sophisticated and effective in response to parallel advances in virus technology. While it is mathematically impossible to identify the presence of a virus with 100 percent reliability, state-of-the-art technology has achieved a close to perfect detection rate of known viruses, and a detection rate perhaps as high as 90 percent, and growing of unknown virus strains. State-of-the-art heuristic virus scanners, for instance, are capable of detecting at least 70-80 percent of *unknown* viruses.[103]

¶140 Organizations, such as Virus Bulletin, West Coast Labs and others, periodically publish evaluations of commercial anti-virus products. Virus Bulletin,[104] an industry leader, uses a recently updated database of virus strains to test anti-virus software for their so-called "100 percent award." Products receive this award if they successfully detect all the strains in the database, suggesting that they are capable of detecting virtually all known strains. Anti-virus software that has consistently made this grade includes products such as Norton AntiVirus, Sophos Anti-Virus and VirusScan.[105]

¶141 West Coast Labs[106] evaluates anti-virus software as much for its detection ability as for its ability to eliminate viruses. Products such as Norton AntiVirus, VirusScan, and F-Secure, among others, have recently been certified for their ability to detect *and eliminate* 100 percent of known virus strains.[107] Other organizations, such as the Virus Test Center at the University of Hamburg,

[103] Carey Nachenberg, *Future Imperfect*, VIRUS BULLETIN, August 1997, at 6-7; Francisco Fernandez, *Heuristic Engines*, Proc. 11th Annual Virus Bulletin Conference, Sept. 2001 (not published); Alex Shipp, *Heuristic Detection of Viruses Within e-Mail*, Proc. 11th Annual Virus Bulletin Conference, Sept. 2001 (not published).

[104] *See* http://www.virusbtn.com.

[105] DUNHAM, *supra* note 1, at 150-151 tbl. 6.3.

[106] *See* http://www.check-mark.com/.

[107] DUNHAM, *supra* note 1, at 154 tbl. 6.6.

regularly test anti-virus software and publish their results, including a list of software with a 100 percent detection rate.[108]

¶142 Some of the most effective anti-virus programs are available free of charge, at least for private users. Free software includes products such as VirusScan, which made Virus Bulletin's 100 percent Award list, and has received similar honors from West Coast Labs. Norton Antivirus, an anti-virus product that has been similarly honored, and which offers additional features such as a user-friendly interface, powerful scan scheduling options, heuristic technology for the detection of unknown strains, and SafeZone quarantine protection, is available, at the time of writing, for a modest price.[109]

¶143 A high detection rate is not limited to known virus strains. State-of-the-art heuristic scanners, such as Symantec's Bloodhound technology and IBM's AntiVirus boot scanner are capable of detecting 70 to 80 percent of unknown viruses.[110] Heuristic technology is relatively inexpensive. Symantec's Bloodhound technology, for instance, is incorporated in the Norton Antivirus product, which (at the time of writing) was available at a modest price.[111]

¶144 The technological trend is towards greater sophistication and effectiveness, and a higher detection rate. The IBM corporation, for instance, a major center of virus research, has recently been awarded a patent for an innovative automatic virus detection system based on neural network technology.[112] The system uses artificial intelligence techniques that mimic the functioning of the human brain to enable it to identify previously unknown virus strains. The neural network is shown examples of infected and uninfected code (e.g. viral and uninfected boot sector samples), and learns to detect suspicious code. Care was taken to minimize the occurrence of false alarms. The system reportedly captured 75 percent of new boot sector viruses that came out since its release, as well as two reports of false positives. Subsequent updates of the product were designed to eliminate the type of false positives that occurred in previous versions.

¶145 Ambitious research programs are underway that augur well for an even greater detection rate. The inventors of the IBM neural network technology view it as a precursor to an immune system for cyberspace that operates analogously to the human immune system. This envisioned cyber immune system will operate through the Internet to "inoculate" users globally to a virus, within minutes of its initial detection.[113]

¶146 Effective prevention technology, capable of eliminating a substantial proportion of errors, is a prerequisite for a high avoidable to unavoidable error ratio. Virus detection technology has achieved this requirement to a high degree. The modest cost of anti-virus software, especially compared to the high danger level inherent in virus infection (the subject of the next subsection), makes these precautions cost-effective and place them within the scope of a duty of due care. Innovative research promises that the trend towards "perfect" detection and elimination will continue, and perhaps accelerate.

C. Danger rate

¶147 An accident with a high danger rate demands a high Learned Hand investment in durable as well as non-durable precautions. This is evident from the expressions derived for x*, y*, and y':

$x^* = (\square AD)^{1/(1\square\square)}$; $y^* = (\square BD)^{1/(1\square\square)}$; $y' = (\square BD)^{1/(1\square\square)} \bullet (1-\square)^{\square/(1\square\square)}$, all increasing in the danger rate, D.

[108] *See* http://agn-www.informatik.uni-hamburg.de/vtc/naveng.htm.

[109] DUNHAM, *supra* note 1, at 158-59.

[110] *See* discussion of heuristic detection technologies in Section III.

[111] DUNHAM, *supra* note 1, at 158-59. *See also* http://www.symantec.com/nav/nav_mac/.

[112] Gerald Tesauro et al., *Neural Networks for Computer Virus Recognition*, IEEE EXPERT, Aug. 1996, at 5-6.

[113] *See generally* J.O. Kephart et al., *Computers and Epidemiology*, 30 IEEE SPECTRUM 5, May 1993, at 20-26 (discussing the mechanics of a computer immune system that would inoculate computers against computer viruses).

¶148 The set of avoidable errors, V(x*, y*) is increasing in (x*, y*). The consequence of a high danger rate is, therefore, a large set of avoidable errors, V(x*, y*), a relatively small set of unavoidable errors, V_U, and a high avoidable-to-unavoidable error ratio, $\frac{\square[V(x^*,\square y^*)]}{\square\square[V_U]}$. The relationship between the danger rate, due care, and the avoidable to unavoidable error ratio is well-recognized by the courts.[114]

¶149 A high danger rate also gives a high *a priori* probability of negligent viral transmission, as is evident from the expression:

Prob(N) = $\square[V(x^*, y^*)] - \square[V(x^*, y')] = B(\square BD)^{\square/(1\square\square)}[1 - (1 - \square)^{\square/(1\square\square)}]$, which is increasing in D.

¶150 A high danger rate therefore strengthens the *res ipsa* inference in two ways. It requires a high level of precaution, resulting in (i) a high avoidable-to-unavoidable error ratio, leaving little room for non-negligently caused errors, and (ii) a high *a priori* probability of negligence, because of the higher likelihood of a compliance error.

¶151 Having established that a high danger rate strengthens a general *res ipsa* case, we now argue that the danger rate specifically associated with computer virus infection is unusually high, both in an absolute sense as well as compared to general computer security hazards and hardware and software errors. Several distinctions make viruses particularly harmful, namely their generality, range of potential harm, persistence, high (and growing) prevalence and impact (organizational, as well as monetary).[115]

¶152 Traditionally, a computer security breach has been related to a particular identifiable weakness in a system, such as a flaw in the operating system (OS). The OS may, for instance, allow unauthorized access to part of the memory to a hacker who invokes the crucial system facilities. Viral infection is a more general security threat, which makes it harder to plan a comprehensive preventative strategy. It can enter the system or network in multiple ways, and any and every program or data file is a potential target. It can be programmed to carry virtually any conceivable resource-dissipating or destructive function, and it can attach to any part of a system.[116]

¶153 The shape and form of viral attacks evolve continuously, as evidenced by the appearance of a progression of stealth, polymorphic, macro and e-mail viruses. Advances in computer technology continuously open up new opportunities for virus writers to exploit. Malevolent software exploiting e-mail technology is a prime example. Conventional wisdom once asserted that it was impossible to contract a virus by simply reading an e-mail message. This wisdom was promptly invalidated by advances in virus technology designed to exploit the unique characteristics, as well as obscure weaknesses and little-known flaws, in new technologies such as JavaScript.

¶154 JavaScript is a language developed by Netscape in collaboration with Sun Microsystems to increase interactivity and control on Internet Web pages, including the capability to manipulate browser windows. The JavaScript e-mail worm, JS.KAK, which appeared at the end of 1999, exploited an obscure Internet Explorer security flaw to disrupt computer systems and destroy data. It infects e-mail attachments and, when the e-mail message is opened, automatically compromises the computer system without having the user open the attachment. A related, but less well-known and shorter-lived e-mail virus, the so-called BubbleBoy, exploited a security hole in the Auto-Preview feature in Microsoft to send a copy of itself to every listing on the user's adress list. BubbleBoy was

[114] *See, e.g.*, Housel v. Pacific Electric Railway Co., 167 Cal. 245, 249 (Cal. 1914) (reasoning that, given the high level of care required of a carrier to its passengers, "a collision would not happen in the ordinary course of events if the carrier exercised such care . . . .").

[115] FREDERICK B. C OHEN, *supra* note 10, at 24-27; LAWRENCE M. BRIDWELL & PETER TRIPPETT, ICSA LABS 6TH ANNUAL COMPUTER VIRUS PREVALENCE SURVEY 2000 6-7 (2000).

[116] FREDERICK B. COHEN, *supra* note 10, at 24 ("The virus spreads without violating any typical protection policy, while it carries any desired attack code to the point of attack. You can think of it as a missile, a general purpose delivery system that can have any warhead you want to put on it. So a virus is a very general means for spreading an attack throughout an entire computer system or network.").

one of the first attachment-resident viruses that did not require the user to open the attachment in order to do its harm.[117]

¶155 The scope of potential harm caused by computer viruses is unprecedented. In a typical conventional security breach, a hacker may access an account, obtain confidential information, and perhaps corrupt or destroy data. The damage could of course be substantial, but it is nevertheless limited to the value of the data contained within the compromised system or network. If, instead, a hacker accesses an account by releasing a virus into the system, the virus may spread across computers and networks, even to those not physically connected to the originally infected system.[118] Whereas the conventional hacker can destroy data worth, say an amount D, releasing a virus to do the same job can cause this harm several times over by spreading into N systems, causing damage of magnitude N•D, where N can be very large. Although the two types of security breaches do similar damage in a particular computer, the virus' greater danger rate lies in the fact that it can multiply and repeat the destruction several times over.[119]

¶156 Dr. Fred Cohen provides a dramatic illustration: "[s]itting at my Unix-based computer in Hudson, Ohio, I could launch a virus and reasonably expect it to spread through 40% of the Unix-based computers in the world in a matter of days. That's dramatically different from what we were dealing with before viruses."[120] A worm (the so-called "Morris Worm") designed and released by a Cornell University student, Robert T. Morris, effectively shut down the Internet and other networks connected to it.[121] It was not designed to damage any data, but conservative estimates of the loss in computer resources and availability range between $10 million and $25 million.[122]

¶157 A third major distinction that gives viruses a higher danger rate than general security hazards is their persistence. A virus can never really be entirely eliminated from a system. Generally, when a bug is fixed or a security flaw plugged, that problem can be considered eliminated from the particular system. In the case of viruses, however, one can never be sure that a particular virus will never return to the system. An infected program may be deleted and restored from a backup, but the backup may have been made after the backed-up program was infected and thus may contain a copy of the virus. Restoring the program will then also restore the virus. This may happen, for instance, in the case of a virus that lies dormant for a while. During its dormancy, periodic backups will also back up the virus. When the virus becomes active, deleting the infected program and restoring it

[117] ROGER A. GRIMES, MALICIOUS MOBILE CODE 394 (2001).

[118] *See, e.g.*, Robin A. Brooke, *Deterring the Spread of Viruses Online: Can Tort Law Tighten the "Net"?*, 17 REV. LITIG. 343, 361 (1998) ("The market now provides enough statistics indicating both high risk and potentially widespread damage from virus attacks, while either programming prevention or off-the-shelf capabilities to detect viruses may impose a proportionally smaller burden."); *Id.* at 348 ("Widespread proliferation of a virus originally undetectable becomes compounded very quickly. Independent actors along the transmission chain can be unaware of malevolent software residing in their computer, network, files, or disks, even if they use virus protection software, because the software may not sufficiently detect more sophisticated code."). *See also*, ALLAN LUNDELL, VIRUS!, at vii (1989) ("Most mainframe computers can be successfully subverted within an hour. Huge international networks with thousands of computers can be opened up to an illicit intruder within days." (Quoting Dr. Fred Cohen)); HRUSKA, *supra* note 28, at 13 ("[N]ew viruses are highly destructive, programmed to format hard disks, destroy and corrupt data. As viral infections become more and more widespread, the danger of damage to data is increasing at an alarming pace.); *Id,* at 13-14 ("The virus danger is here to stay. In the USA, the Far East and Africa it has already reached epidemic proportions . . . In just three months in the Spring of 1989, the number of separately identifiable viruses increased from seven to seventeen.").

[119] DUNHAM, *supra* note 1, at xx ("Just one virus infection can erase the contents of a drive, corrupt important files, or shut down a network.").

[120] FREDERICK B. COHEN, *supra* note 10, at 25. *See also* Gringras, *supra* note 2, at 58 ("A computer harbouring a virus can, in a matter of hours, spread across continents, damaging data and programs without reprieve."); Bradley S. Davis, *It's Virus Season Again, Has Your Computer Been Vaccinated? A Survey of Computer Crime Legislation as a Response to Malevolent Software*, 72 WASHINGTON LAW QUARTERLY 379, 437 n.225 (1994) ("[A] user whose computer was infected could connect to an international network such as the Internet and upload a file onto the network that contained a strain of malevolent software. If the software was not detected by a scanning system . . . on the host computer, infection could spread throughout the Internet through this simple exchange of data."; PHILIP FRITES ET AL., *supra* note 1, at 22.

[121] *See* ROGUE PROGRAMS: VIRUSES, WORMS, TROJAN HORSES 203 (J. HOFFMAN ed., 1990), for an account of the "Internet Worm Incident."

[122] PHILIP FRITES ET AL., *supra* note at 51, 52.

from the backup will only repeat the cycle.[123] Even if the backup is not contaminated, any user of the system with an infected floppy disk could reintroduce the virus into the disinfected system.[124]

¶158 Many virus strains tend to survive progressively new generations of software. Replacing an old, infected spreadsheet program with a new clean version will temporarily eliminate the virus, but the new version will not be immune to that particular virus. If the virus makes its way back, e.g. through a user's infected floppy, it will eventually re-infect the new program.[125]

¶159 Proximate cause considerations may limit damages to primary damage, i.e. harm resulting from direct contact with the infected software.[126] But when defendants are controllers of ftp sites or web sites, they will have many plaintiffs who suffer primary damage.[127] Every visitor to the site, for instance, could potentially be directly infected by malicious code residing on the site and suffer primary damage.

¶160 The high danger rate associated with computer viruses makes them a potentially potent and destructive tool for a perpetrator of terrorism, industrial espionage or white-collar crime.[128] U.S. security agencies are reportedly experimenting with the use of malicious software as a strategic weapon,[129] and the Pentagon established a SWAT team, administered by the Computer Emergency Response Team Coordination Center, designed to combat destructive programs such as the Morris Worm.[130]

¶161 A recent comprehensive survey organized by ICSA Content Security Lab reports rapidly growing virus prevalence and damage.[131] Virtually all of the companies surveyed had experienced at least one virus encounter during the two year survey period (1998 through early 2000). The escalation of virus encounters is illustrated in the table below.[132] The table lists the average monthly infection rate per 1,000 PCs for the first two months of years 1996 through 2000. Regression

[123] Shane Coursen, *How Much is that Virus in the Window*, VIRUS BULLETIN 1996, 15 (describing a common virus named Ripper, that slowly modifies data while the data is being archived, resulting in corrupted backups.); KEN DUNHAM, BIGELOW'S VIRUS TROUBLESHOOTING POCKET REFERENCE, (McGraw-Hill 2000), at 129-130.

[124] Robin A. Brooke, Deterring the Spread of Viruses Online: Can Tort Law Tighten the "Net"?, 17 REV. LITIG. 343, 362 n. 95 ("It is likely impossible to eradicate viruses completely. Simply disinfecting a computer system could cost a staggering amount. In 1990, computer infection in the United States alone was estimated to be one percent, or about 500,000 computers . . . [u]nfortunately, even having a virus removed provides no guarantee of safety from further virus harm. In the United States, 90 percent of all infected users experience re-infection within 30 days of having the original virus removed."); Shane Coursen, *How Much is that Virus in the Window*, VIRUS BULLETIN 1996, 13 ("[T]he fix must be implemented in such a way that it is all-encompassing and simultaneous across infected sites. Tending to one site and neglecting another will surely allow a persistent virus to work its way back again."]; *Id.* at16 ("Cleaning your program of a virus does not guarantee that it will not come by for another visit. Just one leftover diskette or program can have a snowball effect and start another virus outbreak. Within a matter of hours, the entire business could be under siege again. Any time spent cleaning up from the initial infection or outbreak can easily be lost in those few hours. The complete virus recovery process would have to be repeated.").

[125] *See, e.g.*, COHEN, *supra* note 10, at 27 ("Eventually you probably change every piece of software in your computer system, but the virus may still persist. When you go from DOS 2.01 to DOS 2.3, to 3.0, to 3.1 to 3.2 to 4.0 to 4.1 to 5.0 to 6.0 to OS/2, the same viruses that worked on DOS 2.01 almost certainly works on each of these updated operating systems. In fact, if you wrote a computer virus for the IBM 360 in 1965, chance[s] are it would run on every IBM-compatible mainframe computer today, because these computers are upwardly compatible."). Some viruses do become extinct over time, however. *See, e.g.,* DUNHAM, *supra* note 1, at xii ("[M]any older Macintosh viruses do not function correctly on System 7.0 or later. On PCs, many DOS file-infecting viruses are no longer as functional or successful in the Windows operating system. Still, older viruses continue to work on older operating systems and remain a threat for users of older systems.").

[126] CLIVE GRINGRAS, THE LAWS OF THE INTERNET 61 (Butterworths, 1997).

[127] A File Transfer Protocol, or ftp, is a language that allows files to be transferred between computers. *See id.* at 7.

[128] *See*, FITES, *supra* note 1, at 63-65 (describing the use of viruses to perpetrate acts of sabotage, terrorism, and industrial espionage); COHEN, supra note 10, at 151-152. Stoll, Stalking the Wily Hacker, 31 Comms. ACM484 (1988).

[129] Jay Peterzell, *Spying and Sabotage by Computer: The United States and Its Adversaries are Taping Databases—And Spreading Viruses,* TIME MAGAZINE, Mar. 20, 1989, at 25.

[130] ROGUE PROGRAMS: VIRUSES, WORMS, AND TROJAN HORSES, *supra* note 24, at 92 n. 133.

[131] BRIDWELL & TIPPETT, *supra* note 115, at 50-51 ("The virus risk continues to get worse despite corporate efforts . . . The likelihood of a company experiencing a computer virus has approximately doubled for each of the past five years. This is the case in both infection rates and costs . . . Computer viruses were not only more prevalent in surveyed corporations, they were more costly, more destructive, and caused more real damage to data and systems than in the past. This is all true despite the increased use of anti-virus products in more places and despite more aggressive updates and management of such products.").

[132] BRIDWELL & TIPPETT, *supra* note 115, at 10.

analysis of the data estimates an annual growth rate of 22 virus encounters per 1,000 PCs per month.[133]

Year	**Monthly Infection Rate per 1,000 PCs**
1996	10
1997	21
1998	32
1999	80
2000	91

Respondents report productivity losses, corrupted files and lost data, unavailability of computing resources, and loss of user confidence as the major harmful effects of virus attacks. The survey reported a median server downtime of 21 hours and a median of 7 lost person days after a virus attack. Respondents estimated the direct dollar cost of their most recent disastrous attack at $120K, with one estimate as high as $2 million.[134] Conservative estimates of damage from the Morris worm run as high as $25 million.[135]

¶162 The Learned Hand formula balances precautionary costs against expected harm; the latter is the product of the probability of infection and a measure of the damage resulting from infection. We have argued that one side of the equation, precautionary costs, are modest; effective technology, such as Norton AntiVirus, sells for only a modest fee and is capable of detecting virtually 100 percent of known strains and most unknown strains. The two terms that determine expected harm, prevalence and damage, are both significant. The Learned Hand formula applied to anti-virus precautions suggests that the high danger level of virus infection, the high (and rapidly increasing) prevalence of viruses, and the modest cost of anti-virus precautions create a legal duty to implement sophisticated and effective anti-virus technology that is capable of avoiding a large proportion of all virus strains. The result is a high avoidable-to-unavoidable error ratio, and a strong *res ipsa* case.

¶163 We now turn to the second major factor that creates a strong *res ipsa* case, namely a high *a priori* probability of negligent viral transmission.

D. Virus infection and compliance error

¶164 The mathematical model developed in section IV predicts that a rational software or service provider will take *durable* anti-viral precautions at the required due care level, denoted x*. However, the provider's *non-durable* precautions (denoted y') may fall below the due care level (y*), due to efficient non-compliance (i.e. y' < y*).

¶165 The difference between y' and y*, $y^* - y' = [1 - (1-□)^{□/(1□□)}]\bullet y^*$, is increasing in y*. Furthermore, the resulting *a priori* probability of negligence,

¶166 $\text{Prob}(N) = □[V(x^*, y^*)] - □[V(x^*, y')] \ = \ B(y^*)^{□}[1 - (1 - □)^{□/(1□□)}]$, is likewise increasing in y*. This formalizes the intuition that durable precautions complemented by high rates and intense levels of non-durable precautions (a high y*) create a high probability of a compliance error. This result suggests that most negligent virus transmission will be due to compliance errors. The analysis of virus detection technology in this subsection suggests that compliance errors are likely, resulting in a high *a priori* probability of negligent viral transmission.

[133] BRIDWELL & TIPPETT, *supra* note 115, at 10-11.

[134] *See generally* BRIDWELL, *supra* note 115, at 6, 9 (distinguishing between an "encounter"—where a virus is detected before it causes significant harm in an organization—and a "disaster" where the virus infects more than 25 machines or files).

[135] *See* DAVID HARLEY ET AL., VIRUSES REVEALED 347-52 (Osborne/McGraw-Hill 2001).

¶167 Technical defenses against computer viruses consist of a durable precaution, complemented by essential non-durable precautions.[136] Durable anti-virus precautions come in four main categories: pattern scanners, activity monitors, integrity monitors, and heuristic scanners.[137] An activity monitor is an example of a non-durable complementary precaution that halts execution or issues a warning when it senses virus-like behavior. This non-durable precaution requires human intervention, consisting of observation and interpretation of monitor alerts and appropriate response.

¶168 Virus scanners operate by searching for virus patterns in executable code and alerting the user when an observed pattern matches a virus signature stored in a signature database. Non-durable precautions complementary to a scanner include regular maintenance and updating of the virus signature databases, monitoring scanner output and responding to a pattern match. An inadequately maintained signature database would reduce the effectiveness of a scanner, and virus alarms are worthless if ignored.

¶169 Several factors make compliance burdensome. Integrity checkers and heuristic scanners produce fewer false negatives but far more false positives than regular scanners. A large number of false positives makes compliance more burdensome and efficient lapses more likely. False positives tend to undermine confidence in the anti-virus strategy, perhaps to the point of diminishing effectiveness. If the probability of a false alarm were high enough, it might be rational and efficient for a human operator to ignore some alarms. An ignored alarm may turn out to be real and may result in the transmission of a virus. If the Learned Hand precautionary level required attention to all alerts, the courts would view such a lapse as negligence, even if the compliance error were efficient from the viewpoint of the human operator.

¶170 Scanners require a frequently updated viral pattern database since new viruses are discovered at a high rate. For instance, IBM's High Integrity Computing Laboratory reported that by June 1991 new signatures were added to their collection at the rate of 0.6 per day. By June 1994 this rate had quadrupled to 2.4 per day, and has since quadrupled yet again to more than 10 a day.[138] By the Learned Hand formula, the high danger rate associated with viral infection imposes demanding non-durable precautions, such as a high frequency database updating and diligent monitoring of, and response to, all alarms, regardless of the frequency of prior false alarms. Some critical applications may require virtually continuous updating, incorporating new virus strains in real time as they are discovered.

¶171 In applications where the danger rate is particularly high, the durable precautionary demands will be commensurately stringent. Critical applications may require precaution levels, such as a sophisticated AI-based scanning technology, multiple scanners instead of a single scanner, multiple detection techniques such as combined integrity checking and scanning, and scanning of all locations instead of only those locations most likely to be infected.

¶172 The high durable precaution level demanded by such critical applications must be complemented by high non-durable compliance rates in order to be effective. For example, use of multiple detection techniques multiplies the number of warnings and other outputs to manage. More intense detection produces not only more warnings, but also more false alarms. As compliance becomes more difficult, the cost of perfect consistency increases rapidly, eventually outweighing the benefits of perfect consistency, and resulting in less compliance. Such compliance errors would be

[136] *See generally* COHEN, *supra* note 10, at 148 (emphasizing the importance of non-durable precautions in an anti-viral strategy: "Suppose we want to protect our house from water damage. It doesn't matter how good a roof we buy . . . We have to maintain the roof to keep the water out. It's the same with protecting information systems.").

[137] *See infra* Part III.

[138] *See* Jeffrey O. Kephart et al., *Automatic Extraction of Computer Virus Signatures*, 4TH VIRUS BULLETIN INTERNATIONAL CONFERENCE, 179-194, at §1 (1994). *See also* Jennifer Sullivan, *IBM Takes Macro Viruses to the Cleaners*, WIRED NEWS, Dec. 4, 1997 ("It is estimated that 10 to 15 new Word macro viruses . . . are discovered each day; DUNHAM, *supra* note 1, at xix ("[A]n estimated 5 to 10 new viruses are discovered daily, and this number is increasing over time."). In the late 1990's, new viruses were discovered at the rate of 8 to 10 per day. Steve R. White et al., *Anatomy of a Commercial-Grade Immune System*, IBM Thomas J. Watson Research Center research paper, at http://www.research.ibm.com/antivirus/SciPapers/White/Anatomy/anatomy.html.

efficient from the viewpoint of a software provider, who bears the cost of consistent compliance. It would nevertheless be regarded as negligence by the courts that do not consider such costs.[139]

¶173 In conclusion, anti-virus technology consists of a durable component, complemented by high rates and intense levels of non-durable precautions. The analytical results in this article show that the high cost of perfect compliance will result in a high expected compliance error rate that courts will interpret as negligence. The result is a high *a priori* probability of negligent viral transmission, Prob(N), and a strong *res ipsa* case.

E. Role of accident signature

¶174 Accidents sometimes leave evidentiary deposits or "signatures." These signatures may (i) provide a clue to the nature and probable cause of the accident and whether it was avoidable or unavoidable, and (ii) help identify untaken precautions that could have prevented the accident. In *City of Piqua v. Morris*,[140] for instance, the accident signature was a burst reservoir. This enabled the plaintiff to point to a specific untaken precaution that may have prevented the accident (namely unclogging the reservoir's overflow wickets).[141] In *Ybarra v. Spangard* ,[142] the plaintiff underwent an appendectomy and woke up with a sharp pain in his right shoulder. His shoulder muscles eventually atrophied and became paralyzed. In a subsequent lawsuit against the doctors and hospital staff connected with the operation, the court applied *res ipsa loquitur* against the defendants, reasoning that unconscious patients should rarely experience injuries to the parts of their bodies not receiving treatment if hospital employees are reasonably careful. In this case, the signature of the plaintiff's injured shoulder indicated an injury unrelated to his surgery, hence, in the view of the court, an injury probably the result of negligence.

¶175 The *context* in which an accident occurred and the *technology* involved often dictate the strength of the accident as a *res ipsa* case.[143] A plaintiff injured by a jolt, for example, would be substantially more likely to succeed on a *res ipsa* theory if the jolt had taken place in an elevator rather than in an airplane. Courts tend to deny recovery in injury cases where airplane jolts are caused by air turbulence. For example, in *Kohler v. Aspen Airways, Inc.,* the plaintiff suffered injuries when the plane encountered turbulence and dropped sharply. In a subsequent lawsuit against the airline, the court declined to send the case to the jury on a *res ipsa* theory.[144]

¶176 Airplane jolts often occur without any negligence on the part of the pilot or crew. While there is much that can be done to prevent a crash, little can be done to prevent a jolt. A device that could warn a pilot of approaching wind shears would make many jolt injuries avoidable. The avoidable-to-unavoidable error ratio of such injuries would increase as a result of the device, making jolt injuries stronger *res ipsa* candidates. Such technology, however, does not currently exist commercially.

¶177 In elevator jolt cases, on the other hand, courts are much more likely to allow recovery on a *res ipsa* theory. As a result of technological refinements, accidents are rare in well maintained elevators. Hence, elevator accidents, including jolts, have a high avoidable-to-unavoidable error ratio.[145]

[139] *See* Higginbotham v. Mobil Oil Corp., 545 F.2d 422, 429 (5th Cir. 1977) (noting that advanced technology enhances applicability of *res ipsa*), *cert. denied*, 434 U.S. 830 (1977); Williams v. United States, 218 F.2d 473, 476 (5th Cir. 1955); Weiss v. Axler, 328 P.2d 88, 91 (Colo. 1958) ("[T]he more intensified and diversified our industrialism, mechanics and science become, the more technology and automation advance, the more the doctrine of res ipsa loquitur should take on a stellar role in the law of negligence.").

[140] City of Piqua v. Morris, 120 N.E. 300 (Ohio 1918).

[141] But the court found that even if the untaken precaution had in fact been taken, it would not have been sufficient to prevent the accident and, hence, the defendant was not liable due to lack of causation. *Id.* at 302-303.

[142] 154 P.2d 687 (Cal. 1944).

[143] *See* Grady, *supra* note 15, at 929 ("Accidents take their signatures from the context of the technology involved.").

[144] *Kohler v. Aspen Airways, Inc*, 214 Cal. Rptr. 720 (Ct. App. 1985). *See also*, Kelly v. American Airlines, Inc., 508 F.2d 1379, 1380 (5th Cir. 1975); Gafford v. Trans-Texas Airways, 299 F.2d 60, 61-62 (6th Cir. 1962).

[145] *See, e.g.*, Conerly v. Liptzen, 199 N.W.2d 833, 838 (Mich. Ct. App. 1972). *See also*, *Belding v. St. Louis Pub. Serv. Co.*, 215 S.W.2d 506, 511 (Mo. 1948) (applying doctrine of *res ipsa loquitur* to abrupt halt of a bus); *Redmon v. Metropolitan St. Ry.*, 84 S.W. 26, 29 (Mo. 1904) (involving jolts on a streetcar with circumstantial evidence of negligence).

Furthermore, elevator maintenance requires significant non-durable precautions, such as frequent inspections, so that an accident involving an elevator is likely due to negligence.[146]

¶178 The success of a plaintiff in a case involving a computer malfunction, such as a system crash, likewise depends on context and technology. In the absence of evidence of its cause, a system crash would have to be considered a generic computer malfunction, with a commensurately low avoidable-to-unavoidable error ratio, and thus a weak *res ipsa* case. If the culprit could be correctly identified as a computer virus, the associated context and technology would make it a strong *res ipsa* case, based on the theory developed in this article. Identification of the context and technology is trivial in a jolt-related accident, but much more subtle in a case involving computer malfunction.

¶179 A computer virus carries several identifying features, such as a hexadecimal digital signature, soft information in the viral source code, virus-specific side-effects, and generic side-effects.[147] Although the digital signature is the most reliable of these indicators, the others can, nevertheless, significantly strengthen a virus attack *res ipsa* case.

¶180 The most reliable technical indicator of the presence of a known virus in a computer system is a digital signature, which consists of patterns of hexadecimal digits embedded in the viral code.[148] These signatures are created by human experts, such as researchers at institutions such as IBM's High Integrity Computing Laboratory, who scrutinize viral code and extract sections of code with unusual patterns. The selected byte patterns then constitute the signature of the virus.[149] The IBM High Integrity Computing Laboratory has developed an optimal statistical signature extraction technique that examines all sections of code in a virus, and selects the byte strings with the lowest probability of occurrence in legitimate code.[150] The ideal virus signature gives neither false negatives nor false positives. In other words, it should ideally always identify the virus when present and never give a false alarm when it is not.[151]

¶181 The presence of viruses in a computer system may be detected by searching for digital signatures in locations where viruses are likely to reside, such as executable files, boot records or memory. The probability that any of these patterns will be found in uninfected code is small, since they are chosen to minimize occurrence in legitimate code. Digital signatures are therefore reliable indicators of the identity of the virus in which they reside.

¶182 Several observable side-effects of viral infection provide additional clues to the presence and identity of a virus. Side-effects range from relatively harmless annoyances, such as a humorous message or images, to destruction of data and programs. Some of these side-effects are unique to a particular type of virus and helpful in identifying the strain involved.[152] The original version of the so-called Cascade virus,[153] for example, produced a "falling characters" display when a predetermined system date was triggered. The Disk Killer virus, when triggered, displayed a message containing its trademark, the words "Disk Killer."[154] The Italian Ping-Pong virus displays a bouncing ball, [155] and

[146] Grady, *supra* note 15, at 930.

[147] HRUSKA, *supra* note 28, at 56-61.

[148] HRUSKA, *supra* note 28, at 42. *See also*, ROGER A. GRIMES, MALICIOUS MOBILE CODE 449 (2001) ("[W]hile no scanner can detect 100 percent of all malicious programs, a good scanner should be able to detect 100 percent of the common types . . . and over 90 percent of all malicious code types.").

[149] KEPHART ET AL., *supra* note 138, at 179-194, §2.

[150] KEPHART ET AL., *supra* note 138, at 179-194.

[151] KEPHART ET AL., *supra* note 1, at §3.3.5 ("[A] signature extractor must select a virus signature carefully to avoid both false negatives and false positives. That is, the signature must be found in every instance of the virus, and must almost never occur in uninfected programs."). False positives have reportedly triggered a lawsuit by a software vendor, who felt falsely accused, against an anti-virus software vendor. *See generally* KEPHART ET AL., *supra* note 1, at §3.3.5; HRUSKA, *supra* note 28, at 42. *See, e.g.,* HRUSKA, *supra* note 28, at 43-52 (short descriptions and hexadecimal patterns of selected known viruses).

[152] HRUSKA, *supra* note 28, at 40.

[153] HRUSKA, *supra* note 28, at 43.

[154] HRUSKA, *supra* note 28, at 44-45.

[155] HRUSKA, *supra* note 28, at 45.

the destructive South African Friday the 13th virus deleted every file run on a day with the cursed designation.[156]

¶183 Other effects are generic and common to all viral infection cases. Generic side-effects include: unusually long program loading times, errant disk lights, changed interrupt vectors, unaccounted use of random access memory (RAM)[157], decreased available disk space due to the viral code being copied to program files and disk, variation in program size, unusual error messages, inexplicable system crashes, and reduction in free memory space.[158] System services sometimes also behave oddly in the presence of viral infection. Viruses may interfere with or alter system service requests, causing erratic behavior, such as garbled text or images sent to the screen or printer, or failed access to disks.[159] Many of these symptoms may, of course, also be due to a hardware or software bug unrelated to viral infection.[160]

¶184 Identification of a virus as the cause of a computer malfunction, transforms the malfunction from generic to one caused by a virus—and from a malfunction with a low avoidable-to-unavoidable error ratio, to one with a high ratio. However, this transformation depends on the evidentiary reliability of the identification. A digital signature is the most reliable indicator of the presence of a particular viral strain. The virus signature, coupled with strain-specific observable effects, will likely satisfy the courts' preponderance of the evidence standard. Generic virus effects, however, are often indistinguishable from symptoms of a generic computer malfunction, and by themselves are unlikely to satisfy the preponderance of the evidence standard.

F. Summary

¶185 Unique aspects of virus technology, such as a digital signature, the high danger rate associated with virus infection, the structure and operation of anti-viral technologies, and the law and economics of user compliance with non-durable anti-viral precautions, suggest (i) a high likelihood of compliance error, i.e. negligence, and (ii) a large avoidable-to-unavoidable error ratio. These factors create a strong *res ipsa* inference of negligence.

VI. LEGAL AND ECONOMIC ASPECTS OF DAMAGES

¶186 A negligence theory of liability would be irrelevant if no damages were recoverable. A doctrine in tort law, the so-called economic loss rule, appears to significantly limit recovery for damages caused by virus infection. The doctrine denies a defendant's liability for pure economic loss, i.e. loss not based on physical harm to person or property.

¶187 This section analyzes the nature of damages from virus infection and their recoverability in light of the economic loss rule. We conclude that such damages are likely to be recoverable, the economic loss rule notwithstanding, because (i) a virus may cause physical harm due to the malfunction of a computer system, in applications such as medical systems and aviation; (ii) a minority of jurisdictions have relaxed the rule against recovery for pure economic loss; and (iii) an increasing number, perhaps a majority, of jurisdictions recognize electronic information as legally protected property.

A. A damages model

¶188 Damages for virus infection can be classified into two broad categories, namely pre-infection and post-infection damages.[161] Pre-infection damages include the cost of detecting, tracing,

[156] HRUSKA, *supra* note 28, at 47.

[157] The most commonly used computer memory chips.

[158] HRUSKA, *supra* note 28, at 67; ROGUE PROGRAMS: VIRUSES, WORMS, TROJAN HORSES, *supra* note 24, at 42.

[159] ROGUE PROGRAMS: VIRUSES, WORMS, TROJAN HORSES, *supra* note 24, at 42.

[160] FITES ET AL., *supra* note 3, at 100 ("Of course, any of these things can be caused by . . . pressing the wrong keys or by program bugs. These symptoms don't necessarily mean that you have a virus.").

[161] David Hurley, *Nine Tenths of the Iceberg*, VIRUS BULLETIN, Oct. 1999, at 12.

identifying and removing a virus before it enters the system or network. Typical expenses are personnel and managerial expenditures associated with the implementation and maintenance of software that detects a virus automatically at the point of entry, tracing the source of the virus, advising the source, logging the incident, and communicating with the owner of the system on which the incident occurred.

¶189 Post-infection damages can be classified into two main categories, (i) impact on the computing environment resulting from the mere presence of a virus, before execution of the payload; and (ii) damage caused by execution of the payload.

¶190 Viruses modify the computing environment when they install their code on a host program and overwrite or displace legitimate code. Partly overwritten systems programs may become dysfunctional. Macro viruses, for instance, often disable menu options of Microsoft Word. Viral invasion of space in main memory and hard disk may result in impaired performance and the disabling of some programs, including time-critical processes and resource-intensive software. In the absence of virus detection software, these modifications are often unobservable until execution of the payload.[162]

¶191 Damage from execution of the payload comes in three categories, namely loss of availability, integrity, and confidentiality of electronic information.[163] Attacks on availability include renaming, deletion, and encryption of files. Attacks on integrity include modification and corruption of data and files and loss of irreplaceable information. Attacks on confidentiality include security compromises, such as capturing and forwarding passwords, e-mail addresses, and other confidential files.

¶192 Consequential, or secondary, damages include (i) damage (both pre- and post-infection) to systems to which the virus spreads; (ii) damage due to an inappropriate response, such as unnecessarily destroying infected files which could be cheaply disinfected and restored; and (iii) other indirect damages, such as bad publicity and loss of employee morale; the cost of cleanup and disinfection, the cost of restoration of the computer system, and impaired data.[164]

¶193 No viruses have been known to cause damage to hardware (at least at the time of writing), and losses are usually limited to the destruction of data and related direct and indirect costs. We now turn to an analysis of the economic loss rule and its effect on the viability of a negligence cause of action against a perpetrator of virus transmission.

B. The economic loss rule

¶194 The doctrine in tort law, known as the economic loss rule, denies liability of a defendant whose negligence resulted in purely economic harm, i.e., involving no physical harm to person or property.[165] This doctrine is still a majority rule, even though exceptions have appeared in the common law.[166]

¶195 One such exception came when, in a significant turn of events, a unanimous California Supreme Court repudiated the economic loss doctrine in *J'Aire Corp. v. Gregory*.[167] In *J'Aire*,

[162] *Id.* at 13 ("General incompatibility/de-stabilization issues can manifest themselves in several ways. System software, applications, and utilities display unpredictable behavior due to conflicts with unauthorized memory-resident software. Symptoms include protection errors, parity errors, performance degradation, loss of access to volumes normally mounted and unavailability of data or applications.").

[163] *Id.* at 13.

[164] KEN DUNHAM, BIGELOW'S VIRUS TROUBLESHOOTING POCKET REFERENCE 50 (2000) (A user who receives a virus warning "may shut off the computer incorrectly, potentially damaging files, the operating system, or even hardware components like the hard drive.").

[165] Robins Dry Dock v. Flint, 275 U.S. 303 (1927) (early Supreme Court opinion by Justice Holmes). *See also*, 22 Am Jur 2d *Damages* § 20.

[166] For a statement of the minority rule allowing recovery for pure economic loss see Santor v. A.& M. Karagheusian Inc., 207 A.2d 305 (N.J. 1965).

[167] J'Aire Corp. v. Gregory, 598 P.2d 60 (Cal. 1979). For an article generally favorable to the *J'Aire* holding, see Robert Rabin, *Tort Recovery for Negligently Inflicted Economic Loss: A Reassessment*, 37 STAN. L. REV. 1513 (1985). For a generally unfavorable

defendant building contractor was hired to renovate the air conditioning and heating system and to install insulation in a restaurant. Due to the defendant's delays, the restaurant owner suffered financial losses and filed suit, based on a contract theory and a tort theory. The court ignored the contract theory but upheld plaintiff's tort claim for lost profits. The court cautioned that its allowance for recovery based on pure economic harm is subject to the requirement that the harm be "closely connected with the defendant's conduct," and not be "part of the plaintiff's ordinary business risk."[168] Some intermediate courts in California have followed *J'Aire*, allowing recovery for pure economic loss under a negligence theory,[169] while other California courts have declined to do so.[170]

¶196 In 1985, the New Jersey Supreme Court allowed recovery for negligent infliction of economic harm, in *People Express Airlines v. Consolidated Rail Corp.*[171] Mindful of the liability floodgates, the New Jersey Court stressed that recovery should be limited to an "identifiable class of plaintiffs" who are "particularly foreseeable."[172] Montana, Alaska, and Texas courts have followed California and New Jersey in permitting recovery for certain types of negligent infliction of pure economic harm.[173] Other states that have recognized exceptions to the economic loss rule include Arizona, Arkansas, Illinois, Kansas, Maryland, Oregon, and West Virginia.[174] Repudiation of the economic loss rule, however, is still a minority position, as most courts have denied recovery for pure economic loss, including influential opinions by Judge (now Associate Justice) Stephen Breyer[175] and Judge Richard Posner.[176]

C. Computer data as legally recognized property

¶197 The judiciary has not traditionally recognized the integrity of digital information as a protected property interest under tort law, although the law appears to be adjusting to the information age.[177] Statutes have been enacted granting legal protection to computer data against destruction, altering, and unauthorized access. Statutory protection often includes provisions for civil redress. It is, furthermore, expected that courts will follow the lead of state legislatures in granting legal protection to computer data.[178]

¶198 Federal statutes, state criminal statutes, and the common law have all recognized, in various ways, digital information as a protected property interest. The statutes primarily impose criminal liability, and their adjudication of a legal issue does not summarily transfer to a civil case. However, we are concerned with the legal reasoning of the statutes, their legal basis for characterizing electronic information as a legally recognized property interest, and the significance of their

view, see Gary T. Schwartz, *Economic Loss in American Tort Law: The Examples of J'Aire and of Products Liability*, 23 SAN DIEGO L. REV. 37 (1986).

[168] *J'Aire Corp.*, 598 P.2d at 65-66.

[169] *See, e.g.*, Ales-Peratis Foods Int'l Inc. v. American Can Co., 209 Cal. Rptr. 917 (Cal. Ct. App. 1985); Pisano v. American Leasing, 194 Cal. Rptr. 77, 79 (Cal. Ct. App. 1983).

[170] Blatty v. New York Times Co., 221 Cal. Rptr. 236 (Cal. Ct. App. 1985); Fischl v. Paller and Goldstein, 282 Cal. Rptr. 802 (Cal. Ct. App. 1991).

[171] People Express Airlines v. Consolidated Rail Corp., 495 A.2d 107 (NJ 1985).

[172] *Id.* at 116.

[173] Mattingly v. Sheldon Jackson College, 743 P.2d 356 (Alaska 1987); Hawthorne v. Kober Constr. Co., 640 P.2d 467 (Mont. 1982); Moore v. Can. Comm. Bank, 672 SW2d 324, Tex. App. Houston (14th Dist 1984).

[174] Christopher Scott D'Angelo, *The Economic Loss Doctrine: Saving Contract Warranty Law from Drowning in a Sea of Torts*, 26 U. Tol. L. Rev. 591 app. (Spring 1995).

[175] Barber Lines A/S v. M/V Donau Maru, 764 F.2d 50 (1st Cir. 1985).

[176] Rardin v. T & D Mach. Handling, Inc., 890 F.2d 24 (7th Cir. 1989); Miller v. United States Steel Corp., 902 F.2d 573 (7th Cir. 1990).

[177] *See, e.g.*, Michael Rustad and Lori E. Eisenschnidt, *The Commercial Law of Internet Security*, 10 HIGH TECH. L.J., 213, 254 (1995).

[178] Robin A. Brooke, *Deterring the Spread of Viruses Online: Can Tort Law Tighten the "Net"*?, 17 REV. LITIG. 343, 366 n. 117 (Spring 1998).

recognition of computer data and resources as proper subjects for civil action and damages. This legal characterization would be persuasive in a civil case, even if not authoritative.

¶199 The Computer Fraud and Abuse Act (CFAA)[179] is the principal federal statute governing computer-related abuses, such as the transmission of harmful code. Revisions to the Computer Fraud and Abuse Act[180] provide legal protection to the integrity and availability of computer data, computer systems, information and programs.[181] A 1994 modification of the CFAA specifically criminalizes the transmission of malicious code, such as viruses, with the intent to cause damage to information in a computer or to compromise the availability of a computer system.[182] Legal protection includes civil redress, with remedies of compensatory damages, injunctive, and other equitable relief.[183]

¶200 The Electronic Communications Privacy Act[184], The Economic Espionage Act [185], and the Wire Fraud Act[186] all extend similar protection to electronic data. Violation of the Electronic Communications Privacy Act also imposes civil liability on perpetrators.[187]

¶201 State criminal statutes are increasingly recognizing property rights in computer information, paving the way for civil action. Information stored within a computer or on a disk has, for instance, been recognized as property by several state legislatures, including those of Massachusetts,[188] Alabama,[189] Connecticut, [190] Georgia, [191] Ohio, [192] Montana, [193] New Hampshire, [194] Washington,[195] and Wyoming. [196] The Ohio statute, for instance, states "'[p]roperty' includes, but is not limited to . . . computers, data, computer software, financial instruments associated with computers, other documents associated with computers, or copies of the documents, whether in machine or human readable form . . . ."[197] Virtually every state has enacted a computer crime statute, even though they vary quite widely in how they define fundamental terms, such as 'computer-related abuses' and 'electronic information.'[198]

¶202 Although the common law has traditionally considered harm to computer information as more appropriately within the domain of contract rather than tort law,[199] recent decisions have shown a

[179] 18 U.S.C. § 1030 (1986).

[180] 18 U.S.C. § 1030 (West 2002).

[181] 18 U.S.C. § 1030(a)(5), (e)(8) (West 2002).

[182] 18 U.S.C. § 1030(a)(5)(A) (1994).

[183] 18 U.S.C. § 1030(g) (1998) (providing that "[a]ny person who suffers damage or loss by reason of a violation of this section may maintain a civil action against the violator . . . ."). *See, e.g.*, United States v. Morris, 928 F.2d 504 (2d Cir. 1991) (convicting Robert Morris under § 1030 for releasing the infamous "Morris Worm" on the Internet).

[184] 18 U.S.C. § 2701 (1994).

[185] 18 U.S.C. § 1831 (1996).

[186] 18 U.S.C. § 1343 (1984).

[187] 18 U.S.C. §§2701, 2707 (1994).

[188] MASS. GEN. LAWS ANN. ch. 266, §30(2) (West Supp. 1988) (proscribing larceny of property, specifically including "electronically processed or stored data, either tangible or intangible," as well as "data while in transit . . . ."). *See also* P. G. Guthrie, Annotation, *Computer Programs As Property Subject to Theft*, 18 A.L.R.3d 1121 (West 2002) ("Computer programs are property subject to theft.").

[189] Ala. Code 13A-8-101(1) (Supp. 1992) (categorizing electronic data as intellectual property).

[190] CONN. GEN. STAT. ANN. § 53a-250(11) (1997).

[191] GA. CODE ANN. §§ 16-9-90 to -94 (1992).

[192] OHIO REV. CODE ANN. § 2901.01 (10)(a) (1999).

[193] Montana recognizes as property "electronic impulses, electronically processed or produced data or information . . . computer software or computer programs." MONT. CODE ANN. 45-2-101(59)(k) (1995).

[194] N.H. REV. STAT. ANN. § 638:16 (Michie 1997) ("Property means anything of value, including data.").

[195] WASH. REV. CODE ANN. § 9A.52.110-.130 (West 1988).

[196] Wyoming classifies electronic data as intellectual property and prosecutes crimes involving impairment of computer data as violations of intellectual property rights. WYO. STAT. §§ 6-3-501 to -505 (1988).

[197] OHIO REV. CODE ANN. § 2901.01(A)(10)(a) (1999).

[198] Mark D. Rasch, *Criminal Law and The Internet, in* THE INTERNET AND BUSINESS: A LAWYER'S GUIDE TO THE EMERGING LEGAL ISSUES 141 (Joseph F. Ruh Jr. ed., 1996).

[199] *See* Bonna Lynn Horovits, *Computer Software as a Good under the Uniform Commercial Code: Taking a Byte out of the Intangibility*

greater willingness by the courts to recognize tort damages without a physical injury,[200] including damages resulting from loss of computer use,[201] and intentionally released viruses.[202]

¶203 A New York court, for instance, convicted a defendant of charges, including computer tampering, for installing a logic bomb that crashed a computer system.[203] A court in Texas convicted a defendant of "harmful access to a computer" for employing malevolent code to destroy payroll data.[204] In *In re Brandl,* [205] the plaintiff brought action against a bookkeeper/computer operator for releasing a virus into plaintiff's system, on a theory of intentional interference with existing and prospective business relations. The defendant did not respond and a default judgment was issued in favor of the plaintiff.[206]

¶204 A series of English criminal cases have held that altering a magnetic disk constitutes damage to property. In a leading case, Lord Lane CJ of the Court of Appeal stated that interference with data on a disk that diminished the usefulness of the disk to its owner is sufficient grounds to establish liability for damage to legally protected property.[207] It is significant that the Court's analysis emphasized the tangible effect (diminished usefulness of disk) of the impairment of intangible property (electronic data).

¶205 Viruses that do not destroy or alter data nevertheless consume valuable computing resources. The common law has recognized computing resources as property subject to an actionable tort, such as trespass. Several courts allowed a trespass cause of action in cases where computer-generated and computer-transmitted electronic signals, such as unwanted e-mail, had consumed a target computer's resources. In *CompuServe v. Cyber Promotions,*[208] defendant Cyber Promotions sent unsolicited e-mail advertisements to the subscribers of the plaintiff Compuserve, an Internet service provider. The volume of unsolicited mass mailings placed a significant burden on the processing and storage capacity of CompuServe's equipment and slowed down the transfer of information between computers. CompuServe petitioned the court to enjoin the defendant, arguing that defendant's continued unprivileged transmission of electronic messages to its computer equipment constitutes trespass on its personal property. In granting plaintiff's petition, the court relied on several authorities, including the Restatement (Second) of Torts, which indicates that trespass may be actionable where it harms a legally protected interest.[209] The court's language suggests that computing resources constitute (i) a property interest, (ii) with legal protection against invasions of its integrity, including invasions that diminish the economic interest of the owner of the computing resources and invasions that do not physically diminish the property interest.[210]

Myth, 65 B.U.L. REV. 129 (1985). *See also* John M. Conley, *Tort Theories of Recovery Against Vendors of Defective Software*, 13 RUTGERS COMPUTER & TECH. L.J. 1 (1987).

[200] *See* cases cited under discussion of economic loss rule, *supra.*

[201] *See, e.g.*, CompuServe v. Cyber Promotions, 962 F.Supp 1015 (S.D. Ohio 1997); Thrifty Tel, Inc. v. Bezenek, 46 Cal. App. 4th 1559 (Cal. Ct. App. 1996).

[202] North Tel, Inc. v. Brandl (*In re* Brandl), 179 B.R. 620 (Bankr. D. Minn. 1995).

[203] Werner, Zaroff, Slotnick, Stern & Askenazy v. Lewis, 588 N.Y.S.2d 960, 961 (New York City Civ. Ct. 1992).

[204] Burleson v. State, 802 S.W.2d 429, 432 (Tex. App. 1991); *see, also*, United States v. Riggs, 739 F. Supp. 414 (N.D. Ill. 1990) (holding that stolen software and other computerized information constitute property under the Wire Fraud Act); Ward v. Superior Court, 3 CLSR 206 (Cal. Super. Ct. 1972) (conviction for conversion of proprietary software under trade secret law); United States v. Seidlitz, 589 F.2d 152 (4th Cir. 1978) (computer software constitutes property under the Wire Fraud Act).

[205] *North Tel, Inc.*, *supra*, at 622.

[206] *Id.*

[207] R v. Whiteley 93 Crim. App. R. 25 (Eng. CA 1991); *see also*, Cox v. Riley 83 Crim. App. R. 54 (Eng. CA 1986).

[208] *CompuServe*, 962 F.Supp. 1015.

[209] RESTATEMENT (SECOND) OF TORTS § 218(d) (1965).

[210] *CompuServe*, 962 F.Supp. at 1022 ["To the extent that defendants' multitudinous electronic mailings demand the disk space and drain the processing power of the plaintiffs' computer equipment, those resources are not available to serve CompuServe subscribers. Therefore, the value of that equipment to Compuserve is diminished even though it is not physically damaged by defendants' conduct."]; *see also*, United States v. Sampson, 6 CLSR. 879 (N.D. Cal. 1978) (conviction of a defendant who had appropriated computer time, for embezzlement of government property). Some courts have declined to find a property interest in computing resources. *See, e.g.*, Indiana v. McGraw, 480 N.E. 2d 552 (Ind. 1985); New York v. Weg, (NY Crim. Ct. 1982).

D. Analysis and conclusions

¶206 Damage from computer virus infection can be classified into two broad categories, namely (i) dissipation of computing and administrative resources, and (ii) destruction or corruption of electronic data. In exceptional cases, a computer malfunction due to virus infection may result in physical harm.

¶207 Lost computing and administrative resources, although pure economic losses, may be recoverable under the principles established in cases such as *J'Aire Corp. v. Gregory*[211] and *People Express Airlines v. Consolidated Rail Corp.*[212] *J'Aire* would allow recovery for economic loss that is "closely connected with the defendant's conduct," and not "part of the plaintiff's ordinary business risk," both conditions that are usually satisfied in a typical case involving a virus attack. *People Express* articulated a strong proximate cause limitation, namely a limitation to recovery by an "identifiable class of plaintiffs" who are "particularly foreseeable."[213] The articulation of the *People Express* court may be interpreted to mean that recovery should be limited to plaintiffs who have incurred primary damage, i.e. those to whom the virus was first transmitted, as opposed to secondary damage, incurred by those to whom the virus subsequently spread. Even if proximate cause considerations were to limit recovery to primary damages, defendants such as controllers of ftp sites[214] or web sites would have many plaintiffs who suffer primary damages. Every visitor to the site may be directly infected by malicious code residing on the site and suffer primary damage.

¶208 Alternatively, a plaintiff seeking damages related to virus infection may rely on the common law recognition of computing resources as property subject to an actionable tort such as trespass. Several courts have allowed a trespass cause of action in cases where computer-generated and computer-transmitted electronic signals, such as unwanted e-mail, had consumed a target computer's resources.[215] The analogy can be plausibly extended to computer viruses as "computer-generated and computer-transmitted electronic signals."

¶209 Damage to electronic data and programs may be recovered under *J'Aire*, or by relying on the reasoning in statutes that provide for the legal protection of such data against destruction, altering, and unauthorized access. Although the statutes primarily impose criminal liability, they provide the legal basis and reasoning for characterizing electronic information as a legally recognized property interest. Several of the statutes also provide for civil redress, implicitly recognizing electronic information as a proper subject of tort action and civil remedies, such as damages and injunctive relief.

¶210 In exceptional cases, a virus may cause physical harm. Even though (at the time of writing) no known virus appears capable of directly damaging computer hardware, the malfunction of a computer system due to viral infection may cause physical harm in certain applications.[216] A virus in the computer system of a hospital may shut down a life support system.[217] A data-altering virus in a computer system may generate a misleading warning label for a prescription drug,[218] or cause

[211] J'Aire Corp. v. Gregory, 598 P.2d 60 (Cal. 1979). For an article generally favorable to the *J'Aire* holding, see Robert Rabin, *Tort Recovery for Negligently Inflicted Economic Loss: A Reassessment*, 37 STAN. L. REV. 1513 (1985). For a generally unfavorable view, see Gary T. Schwartz, *Economic Loss in American Tort Law: The Examples of J'Aire and of Products Liability*, 23 SAN DIEGO L. REV. 37 (1986).

[212] People Express Airlines v. Consolidated Rail Corp. 495 A.2d 107 (NJ 1985).

[213] *People Express*, 496 A.2d 107 at 116.

[214] A File Transfer Protocols, or ftp, is a language that allows files to be transferred between computers. *See, e.g.*, CLIVE GRINGRAS, THE LAWS OF THE INTERNET 7 (1997).

[215] *CompuServe*, 962 F.Supp. 1015; Thrifty-Tel, Inc. v. Bezenek, 46 Cal. App. 4th 1559, 1567 (Cal. Ct. App. 1996); State v. McGraw, 480 N.E.2d 552, 554 (Ind. 1985); State v. Riley, 846 P.2d 1365 (Wash 1993).

[216] *See, e.g.,* Douglas E. Phillips, *When Software Fails: Emerging Standards of Vendor Liability Under the Uniform Commercial Code*, 50 BUS. LAW. 151, 155-56 (1994); Evan I. Schwartz, *Trust Me, I'm Your Software*, DISCOVER, May 1996, at 80 (discussing recent software failures that resulted in injuries or deaths).

[217] Laura DiDio, *A Menace to Society: Increasingly Sophisticated - and Destructive - Computer Viruses May Begin to Take Their Toll in Lives as Well as Dollars*, NETWORK WORLD, Feb. 6, 1989, at 71.

[218] Frye v. Medicare-Glaser Corp., 579 N.E.2d 1255 (Ill. App. 1991) (computer-generated label, which gave inadequate

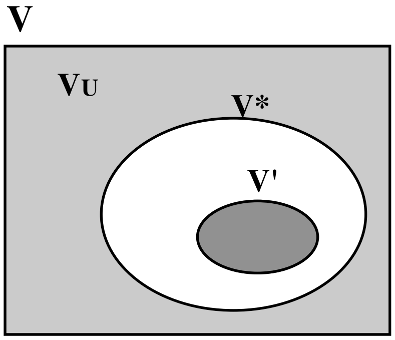
miscalculation of doses of radiation for cancer patients,[219] resulting in harm to health, and perhaps even death. In cases involving physical harm, the economic loss rule does, of course, not limit recovery of damages.

¶211 In conclusion, although the economic loss rule is still a majority rule and may place obstacles to recovery of damages for viral infection, exceptions to the rule have appeared in the common law. Furthermore, federal and state legislatures are responding to the realities of the information age with legal protections for the integrity of electronic data, computer programs, and computer systems. Commentators have contributed to the trend, by arguing in favor of tort liability for virus infection, on public policy grounds.[220] We conclude that victims of malevolent software have the prospect of a significant, and growing, likelihood of successful damages recovery.

VII. CONCLUDING REMARKS

¶212 The victim of a computer virus attack will frequently face insurmountable technical obstacles to directly proving the negligence of the responsible entity. The evidentiary problem can be illustrated as follows. In the diagram below, the rectangle (labeled V) represents the set of all virus strains, known as well as unknown. The software provider has a duty to prevent transmission of the subset represented by the ellipse labeled V*, the avoidable set. The outer colored area, labeled V_U, represents the set of unavoidable strains that even due care would not prevent. The inner dark ellipse, denoted V', represents the set of strains, obviously smaller than V*, that would actually be prevented if the software provider were to fall short of due care.

¶213 If a transmitted virus could be identified as belonging to either V_U or V*, determining liability would be trivial. The defendant would be held liable for transmitting a virus in V*, but acquitted if the virus were in V_U.



¶214 Such identification, however, will not always be possible. A polymorphic virus, for instance, changes its identifying signature each time it infects a new host. Some are programmed to change their signature randomly, implying that the previous signature cannot be ascertained from either the current signature or the structure and logic of the virus. A polymorphic virus, with an "undetectable" signature, may be present in the software at the time the defendant software provider scans for viruses. An undetectable signature is one that has not been commercialized, hence not available in any scanner database. The virus is therefore *unavoidable*. When the virus is transmitted and infects the plaintiff's software, it may change its signature, perhaps to one that is in fact detectable. The current signature would then suggest that the virus is *avoidable*. However, because of the random

warning, was printed out separately from the standard warning and was discarded). Note: this case did not involve viral infection, but a virus could conceivably cause such harm.

[219] Jones v. Minnesota Mining & Manufacturing Co., 669 P.2d 744 (N.M. Ct. App. 1983).

[220] Robin A. Brooke, *Deterring the Spread of Viruses Online: Can Tort Law Tighten the "Net"?,* 17 REV. LITIG. 343, 358 (1998) ["Imposing appropriate tort liability for viral infection should improve market efficiency and confidence in the system by managing realistic business expectations."].

evolution of the signature, the plaintiff cannot prove that it was avoidable at the time of the defendant's scanning. The plaintiff is unable to prove negligence directly, and has to rely on a doctrine such as *res ipsa loquitur*. The main thesis of this article is that virus infection is a strong *res ipsa* case. The intuition of the analytical results can be explained as follows.

¶215 In the diagram above, the white area between V* and V' represents the set of negligently transmitted viruses. A virus strain belonging to this set will be transmitted to the plaintiff, because it is outside of V', while the courts will consider it avoidable (hence, negligently transmitted), because the strain is in V*.

¶216 The larger the "gap" between V* and V', the higher the probability of negligent viral transmission, and the stronger the inference of defendant's negligence. The gap increases as V* increases and V' shrinks. A large V* corresponds to a large avoidable-to-unavoidable error ratio, while a small V' corresponds to a propensity to take precautions below the due care level, i.e. a significant *a priori* probability of negligence. The analysis in this article shows that virus infection is characterized both by a large avoidable-to-unavoidable error ratio, as well as a significant economic incentive to take precautions below due care. The result is a strong inference of negligence and a strong *res ipsa* case.

¶217 The high danger rate associated with computer virus infection plays an important role in the *res ipsa* inference. This aspect becomes crucial when a defendant transmits a virus that foreseeably finds its way into the so-called national critical information infrastructure. The critical information infrastructure (CII) is, broadly speaking, an interrelated system of computer and communication networks that control and coordinate essential infrastructures, such as water supplies, banking and financial services, telecommunications services, and electrical power. It also includes computer networks that coordinate and control military communications and logistics.[221]

¶218 Software providers and distributors whose products have a connection with the CII have to internalize an extremely high danger rate in their anti-virus precautions. The danger rate is due to the high danger rate inherent in virus infection, compounded by (i) the vulnerability and importance of the strategic interests that are serviced by the CII, and (ii) the interconnectedness of its supporting computer networks.[222] The transportation and telecommunications infrastructures, for instance, both depend on the electrical power infrastructure, which in turn depends on a reliable energy supply. The military depends substantially on commercial communications and information networks.[223] A single well-designed computer virus may spread and disrupt electric power networks, paralyze banking systems, and compromise military and civilian communications, all within a very short time frame.[224] Besides the threat to national security, such an incident may lead to loss of life and huge direct economic losses.[225]

¶219 The analysis of this paper, in light of the extremely high danger associated with disruption of the CII, suggests that viral infection of a network in the CII carries a strong inference of liability under *res ipsa loquitur*, perhaps close to strict liability.

[221] The private sector plays a dominant role in the critical information infrastructure. Most infrastructures are owned by the private sector and the Defense Information Systems Agency depends heavily on commercial communication networks. *See, e.g.*, CENTER FOR STRATEGIC AND INTERNATIONAL STUDIES, CYBERCRIME...CYBERTERRORISM...CYBERWARFARE...: AVERTING AN ELECTRONIC WATERLOO at xiv-xv (1998), *cited in* Gregory Grove, *The U.S. Military and Civil Infrastructure Protection: Restrictions and Discretion under the Posse Comitatus Act*, (CISAC Working Paper, Stanford University).

[222] *See, e.g.*, JOINT STAFF DOCUMENT, INFORMATION ASSURANCE DIVISION, INFORMATION WARFARE at 2 (The Defense and national Information Infrastructures "share terrestrial telecommunications networks, a variety of information databases, and satellite communications networks. These infrastructures connect geographically separated forces and span international boundaries.").

[223] *See, e.g.*, Gregory Grove, *supra* note 220, at 3 ("[T]he Defense Information Systems Agency delivers 95 percent of its communications through public commercial lines . . . .").

[224] *See, e.g* ., JOINT STAFF DOCUMENT, INFORMATION ASSURANCE DIVISION, supra note 221, at 2 ("Warfighting information systems are linked through supporting infrastructures, thus exposed to attacks by a broad range of adversaries . . . .").

[225] It has been estimated, for instance, that a five-hour fiber optic ring failure could result in up to $15 million in lost revenue. *See, e.g.*, WAYNE POPE AND JOHN CHAIMBERLAIN, FIBER PREVENTATIVE MAINTENANCE, http://www.cedmagazine.com/pm/97sp/97spd.htm.