EMORY | LAW

Emory Law Journal

Volume 60 | Issue 5

2010

The Hacker's Aegis

Derek E. Bambauer

Oliver Day

Follow this and additional works at: https://scholarlycommons.law.emory.edu/elj

Recommended Citation
Derek E. Bambauer & Oliver Day, *The Hacker's Aegis*, 60 Emory L. J. 1051 (2010).
Available at: https://scholarlycommons.law.emory.edu/elj/vol60/iss5/1

This Article is brought to you for free and open access by the Journals at Emory Law Scholarly Commons. It has been accepted for inclusion in Emory Law Journal by an authorized editor of Emory Law Scholarly Commons. For more information, please contact law-scholarly-commons@emory.edu.

THE HACKER'S AEGIS

Derek E. Bambauer[*]
Oliver Day[**]

ABSTRACT

Intellectual property (IP) law stifles critical research on software security vulnerabilities, placing computer users at risk. Researchers who discover flaws often face IP-based legal threats if they reveal findings to anyone other than the software vendor. This Article argues that the interplay between law and vulnerability data challenges existing scholarship on how intellectual property law should regulate information about improvements on protected works, and suggests weakening, not enhancing, IP protections where infringement is difficult to detect, lucrative, and creates significant negative externalities. It proposes a set of three reforms—"patches," in software terms—to protect security research. Legal reform would create immunity from civil IP liability for researchers who follow "responsible disclosure" rules. Linguistic reform would seek to make the term hacker *less threatening either by recapturing the term's original meaning, or abandoning it. Finally, structural reform would ameliorate failures in the market for software vulnerability data by having a trusted third party act as a voluntary clearinghouse. The Article concludes by describing other areas, such as physical security, where it may be useful to reform how law coordinates IP improvements.*

---

[*] Associate Professor of Law, Brooklyn Law School.
[**] Security Researcher, Akamai.

The authors thank Oded Burger, Jennifer Burgomaster, Jelena Kristic, Brad Reid, and Chris Vidiksis for expert research assistance. Thanks for helpful suggestions and discussion are owed to Miriam Baer, Fred Bloom, Adam Candeub, Mike Carroll, Jennifer Carter-Johnson, Ed Cheng, Jorge Contreras, Dino Dai Zovi, Erik Dykema, Robin Effron, Dave Fagundes, Shubha Ghosh, Jennifer Granick, Dan Guido, Lital Helman, Brian Lee, Dave Levine, Mike Madison, Jason Mazzone, Phil Malone, Liberty McAteer, Bill McGeveran, Maureen O'Rourke, Sean Pager, Gideon Parchomovsky, Zahr Said, Ted Sichelman, Jessica Silbey, Chris Soghoian, Alexander Sotirov, Marketa Trimble, Scott Velez, Jane Yakowitz, Fred Yen, Julie Cromer Young, the Intellectual Property Colloquium at Brooklyn Law School, and the participants in the Seventh Annual Works in Progress Intellectual Property conference. The authors gratefully acknowledge the Dean's Summer Research Stipend Program and President Joan G. Wexler at Brooklyn Law School for financial support. The authors welcome comments at derek.bambauer@brooklaw.edu and oday@fas.harvard.edu.

INTRODUCTION: != BULLETPROOF[1]

Mike Lynn had done the impossible. He had found a way to crack open the operating system on Cisco internet routers, causing them to run his code.[2] Routers were Cisco's most important product—and the backbone of much of the internet—precisely because they had been legendarily immune to such attacks.[3] Lynn, though, had discovered their Achilles' heel. The routers' vulnerability placed a wide swath of internet infrastructure at risk.

Lynn, an experienced security researcher with the firm Internet Security Systems (ISS), followed the protocol of "white hat" hackers, who probe for computer software and hardware flaws with the goal of discovering, not exploiting, them.[4] He reported his findings to Cisco, which dutifully issued a patch to correct the bug.[5] But Cisco—concerned with damaging the invincible image of its products—refused to draw particular attention to the patch, or to press customers to implement it.[6] Lynn, worried by Cisco's decision not to publicize the fix, prepared to give a presentation at the Black Hat hacker conference in Las Vegas that would detail the basic concepts of the bug, but would withhold information about how to exploit it.[7]

Cisco objected, fervently. Employing a range of legal theories from intellectual property law, the company convinced a federal judge to issue a restraining order preventing Lynn from giving his presentation.[8] The company also forced conference organizers to rip the printed version of Lynn's slides

---

[1] In programming languages, *!=* means "not equal to." *See Built-In Types—Python v2.7.1 Documentation*, PYTHON STANDARD LIBR. § 5.3 tbl., http://docs.python.org/library/stdtypes.html#comparisons (last updated May 13, 2011).

[2] Kim Zetter, *Router Flaw Is a Ticking Bomb*, WIRED (Aug. 1, 2005), http://www.wired.com/politics/security/news/2005/08/68365.

[3] Robert Lemos, *Cisco, ISS File Suit Against Rogue Researcher*, SECURITYFOCUS (July 27, 2005), http://www.securityfocus.com/news/11259.

[4] By convention, black hat hackers discover bugs for financial gain or malicious reasons, and gray hat hackers behave either as white hats or black hats, depending on the circumstances. The tripartite division, borrowed from movie Westerns, corresponds roughly to good actors (white hats), bad ones (black hats), and those whose orientation varies (gray hats). *See* THOMAS WILHELM, PROFESSIONAL PENETRATION TESTING: CREATING AND OPERATING A FORMAL HACKING LAB 13–18 (2009).

[5] Robert McMillan, *Black Hat: ISS Researcher Quits Job to Detail Cisco Flaws*, INFOWORLD (July 27, 2005), http://www.infoworld.com/d/security-central/black-hat-iss-researcher-quits-job-detail-cisco-flaws-088.

[6] *See* Zetter, *supra* note 2.

[7] Jennifer Granick, *An Insider's View of 'Ciscogate,'* WIRED (Aug. 5, 2005), http://www.wired.com/science/discoveries/news/2005/08/68435.

[8] *Id.*

out of the conference materials, and to turn over CDs containing a copy of his slideshow.[9]

This Article argues that conflicts such as the one between Lynn and Cisco are both increasingly common and socially harmful. Intellectual property (IP) law stifles the dissemination of critical research on software security vulnerabilities. We argue that IP law's incentive effects are superfluous for these bugs, as security research is an exemplar of "peer production" as conceptualized by Yochai Benkler,[10] Eric von Hippel,[11] and Eric S. Raymond.[12] Researchers hunt bugs for a variety of reasons: intellectual curiosity, ideology, reputation, and occasionally remuneration. For vulnerability research, IP law plays a suppressive rather than a generative function—it blocks or limits whether, and how, hackers share their findings.[13] The suppressive effect is heightened by the fact that researchers can rarely, if ever, obtain IP law protection for their findings or insights. We argue that, much as researchers have hacked software to make it behave unexpectedly and thereby serve their purposes, software vendors have hacked IP law, using it for ends unrelated to its original purpose.

Critically, IP law—like the software it protects—malfunctions here. It enables software firms to suppress information about flaws. It presses researchers to avoid legal risks from public disclosure and to gain financially by offering their findings on the black market rather than through legitimate channels. Software-vulnerability research challenges standard intellectual property scholarship on the regulation of information about improving a protected work or invention. Under current doctrine, someone who possesses information about how to improve a work or invention protected by IP has three options: bargain with the IP owner, seek an improvement patent, or infringe. Contemporary scholarship typically focuses on tuning patent and copyright law to generate optimal incentives and to coordinate improvements. Mark A. Lemley argues that it is unnecessary for inventors to capture the full

---

[9] Bruce Schneier, *Cisco Harasses Security Researcher*, SCHNEIER ON SECURITY (July 29, 2005, 4:35 AM), http://www.schneier.com/blog/archives/2005/07/cisco_harasses.html.

[10] Yochai Benkler, *Coase's Penguin, or, Linux and* The Nature of the Firm, 112 YALE L.J. 369, 375 (2002).

[11] ERIC VON HIPPEL, DEMOCRATIZING INNOVATION 93–97 (2005) [hereinafter VON HIPPEL, DEMOCRATIZING INNOVATION]; ERIC VON HIPPEL, THE SOURCES OF INNOVATION 25–26 (1988) [hereinafter VON HIPPEL, THE SOURCES OF INNOVATION].

[12] ERIC S. RAYMOND, THE CATHEDRAL AND THE BAZAAR 49–53 (Tim O'Reilly ed., rev. ed. 2001).

[13] *See generally* Jonathan L. Zittrain, *The Generative Internet*, 119 HARV. L. REV. 1974 (2006) (describing how the internet's architecture empowers users to generate innovation).

social value of their advances, and that patent law should not set this internalization as a goal.[14] Robert P. Merges and Richard R. Nelson analyze the incentive effects of various standards for setting the scope of a patent,[15] as does Edmund W. Kitch.[16] William M. Landes and Judge Richard A. Posner justify control over improvement information by IP owners as useful in reducing transaction costs.[17] Michael A. Heller and Rebecca S. Eisenberg worry about the problem of holdout costs when multiple parties must bargain over improvements.[18] Paul Goldstein assesses how copyright's derivative works doctrine—particularly indifferent to economics—has created adverse effects on incentives to invest in copyrighted works.[19] Current scholarly wisdom thus presses toward conferring control over improvement information to IP owners.

This Article, in contrast, identifies software security research as a counterexample, where IP owners' strong controls over improvement information are harmful. Security bugs are problematic for three reasons: infringement is (1) difficult to detect, (2) socially harmful due to negative externalities, and (3) lucrative. We argue that IP law should be alert to similar situations and that, counterintuitively, such circumstances require a diminution, not an increase, in IP protections. The Article goes on to suggest additional areas where the channeling effect of legal rules over improvement information may be critically important.

This Article is also the first to propose a set of reforms—"patches," in software terms—to protect socially valuable security research, guide behavior of those searching for vulnerabilities, and channel dissemination of vulnerability data toward legitimate consumers. Other legal scholarship treats intellectual property law as a lost cause. Jennifer Stisa Granick argues compellingly against restrictions on vulnerability disclosures, but focuses on IP

---

[14] Mark A. Lemley, *Property, Intellectual Property, and Free Riding*, 83 TEX. L. REV. 1031 (2005).

[15] Robert P. Merges & Richard R. Nelson, *On the Complex Economics of Patent Scope*, 90 COLUM. L. REV. 839 (1990).

[16] Edmund W. Kitch, *The Nature and Function of the Patent System*, 20 J.L. & ECON. 265 (1977).

[17] WILLIAM M. LANDES & RICHARD A. POSNER, THE ECONOMIC STRUCTURE OF INTELLECTUAL PROPERTY LAW 110–11 (2003).

[18] Michael A. Heller, *The Tragedy of the Anticommons: Property in the Transition from Marx to Markets*, 111 HARV. L. REV. 621, 640–41, 667–77 (1998); Michael A. Heller & Rebecca S. Eisenberg, *Can Patents Deter Innovation? The Anticommons in Biomedical Research*, 280 SCIENCE 698, 700 (1998).

[19] Paul Goldstein, *Derivative Rights and Derivative Works in Copyright*, 30 J. COPYRIGHT SOC'Y U.S.A. 209 (1983).

law solely as a barrier.[20] Peter P. Swire, in assessing incentives for vulnerability disclosure, notes law's role as a barrier to a firm's competitors in creating equivalent software.[21] Susan W. Brenner evaluates the Digital Millennium Copyright Act (DMCA) as a form of information censorship.[22] Bruce H. Kobayashi argues for more extensive intellectual property protection to drive adoption of cybersecurity.[23]

Computer scientists are even more pessimistic. Stephen Bono, Aviel Rubin, Adam Stubblefield, and Matthew Green refer to "security through legality" as a "hopelessly flawed methodology."[24] Tom Cross views efforts to limit hackers' investigations as embracing the view that "ignorance makes you safer."[25] And Paul Graham, who invented Bayesian spam filtering, views copyright as "a threat to the intellectual freedom [hackers] need to do their job," which is to reduce the creation and impact of poorly written software code.[26]

In contrast, this Article seeks to adapt IP law, rather than to abandon it as a tool. It proposes three methods of reform to accomplish this. First, we argue that the security research community should try to shift the largely negative, threatening set of connotations associated with the term *hacker*. If bug hunters cannot reclaim the word's original meaning, they should cede it and employ an alternative.

Second, a voluntary intermediary—a vulnerability clearinghouse—should be established to coordinate contact between vendors and researchers, to document identified bugs, and to track their evolution. The clearinghouse can address key structural flaws in the market for vulnerability information that impede legitimate transactions and push researchers to sell information illicitly.

---

[20] Jennifer Stisa Granick, *The Price of Restricting Vulnerability Publications*, INT'L J. COMM. L. & POL'Y, Spring 2005, at 1, http://www.ijclp.net/files/ijclp_web-doc_10-cy-2004.pdf.

[21] Peter P. Swire, *A Theory of Disclosure for Security and Competitive Reasons: Open Source, Proprietary Software, and Government Systems*, 42 HOUS. L. REV. 1333, 1366 (2006).

[22] Susan W. Brenner, *Complicit Publication: When Should the Dissemination of Ideas and Data Be Criminalized?*, 13 ALB. L.J. SCI. & TECH. 273, 348–56 (2003).

[23] Bruce H. Kobayashi, *An Economic Analysis of the Private and Social Costs of the Provision of Cybersecurity and Other Public Security Goods*, 14 SUP. CT. ECON. REV. 261 (2006).

[24] Stephen Bono et al., *Security Through Legality*, COMM. ACM, June 2006, at 41, 42.

[25] Tom Cross, *Academic Freedom and the Hacker Ethic*, COMM. ACM, June 2006, at 37, 39.

[26] PAUL GRAHAM, HACKERS & PAINTERS 51 (2004).

Finally, the Article proposes regulating researcher behavior in exchange for a shield from IP law. If hackers follow a prescribed course of conduct during their investigations—roughly tracking the "responsible disclosure" model used in the security community—they should be granted immunity from civil[27] intellectual property liability for that research.

The goal of these reforms is to channel disclosures of vulnerability information in legitimate directions. Threats of legal liability may prompt researchers to offer their discoveries on the (lucrative) black market or to withhold research altogether, rather than risking a lawsuit by contacting a vendor or publicizing their findings. The reforms may spur researchers to undertake additional investigations, producing more information about bugs; however, any such benefit, while helpful, is secondary to the useful effects on information distribution.

This Article builds on two underlying normative commitments. First, we believe that the proposed slate of changes—the Hacker's Aegis—holds considerable promise for improving the security and reliability of computer software. Hackers, like the open source software community, create public goods by developing information about software flaws. Second, we seek to defend, and hopefully to reorient the perception of, software security research more generally. The term *hacker*, once a cognomen of approval, has become a term of criticism and even fear. This shift misrepresents researchers' activities and the social value they contribute. By protecting software security research, we hope to change perceptions of it.

Part I of this Article describes the ecosystem in which security researchers operate. Part II catalogs the intellectual property tools available to threaten and control hackers, and suggests what doctrinal patches are needed to protect security research. Part III describes the Article's three proposed reforms to mitigate IP's ill effects in this context. This Article concludes with observations on how its analysis can be applied outside the realm of computer software.

---

[27] For a discussion of criminal IP liability for security research, see *infra* Part III.E.

I. THE SOFTWARE SECURITY ECOSYSTEM

A. The Stakes

Finding security bugs matters. Users face an increasingly hostile internet environment—one where malware, viruses, identity theft, phishing, and denial-of-service attacks are ubiquitous. In the United States, hackers took control of over a million personal computers in the last three months of 2009, adding them to the ten million already infected with rogue code.[28] Security firm Panda Labs tested over twenty-two million computers and found that nearly half (48.35%) were infected with malware.[29] The consequences of suffering a hack or an infection can be significant, as the loss of sensitive personal data due to security breaches has become commonplace. A hack at the University of North Carolina School of Medicine exposed the personal data and medical information of approximately 160,000 mammography patients.[30] The Director of the Federal Bureau of Investigation gave up online banking after nearly falling for a phishing e-mail that appeared to come from his bank.[31] A vulnerability in one of Time Warner Cable's standard cable modem/wi-fi router units allowed hackers to change the device's settings and potentially intercept data sent through it.[32] Viruses can even spread from infected personal computers (PCs) to websites.[33] Vulnerabilities in software code create the weaknesses that hackers exploit.

Bugs not only put an individual's information at risk, they create a threat to other internet users as well. Security flaws present two forms of negative externalities. First, each user whose software is compromised increases the risk to her peers. Computers infected with viruses or malware are often

---

[28] Ellen Nakashima, *China Is World Leader in Hacked Computers, Report Finds*, WASH. POST, Feb. 15, 2010, at A3.

[29] *See* ANTI-PHISHING WORKING GRP., PHISHING ACTIVITY TRENDS REPORT: 3RD QUARTER 2009, at 10 (2009), *available at* http://www.antiphishing.org/reports/apwg_report_Q3_2009.pdf.

[30] Eric Ferreri, *UNC Security Breach Less Severe than Thought*, NEWS & OBSERVER CAMPUS NOTES BLOG (Sept. 30, 2009, 3:13 PM), http://blogs.newsobserver.com/campusnotes/unc-security-breach-less-severe-than-thought (discussing exposure of about 160,000 files and 114,000 Social Security numbers); *Hackers Attack UNC-Based Mammography Database*, UNC HEALTH CARE (Sept. 25, 2009), http://news.unchealthcare.org/som-vital-signs/archives/vital-signs-sept-25-2009/hackers-attack-unc-based-mammography-database.

[31] Elinor Mills, *Wife Bans FBI Head from Online Banking*, CNET INSECURITY COMPLEX (Oct. 7, 2009, 4:07 PM), http://news.cnet.com/8301-27080_3-10370164-245.html.

[32] Kim Zetter, *Time Warner Cable Exposes 65,000 Customer Routers to Remote Hacks*, WIRED THREAT LEVEL (Oct. 20, 2009, 6:20 PM), http://www.wired.com/threatlevel/2009/10/time-warner-cable.

[33] Maxim Weinstein, *Local Malware Causes Infected Websites*, STOPBADWARE BLOG (July 16, 2009), http://blog.stopbadware.org/2009/07/16/local-malware-causes-infected-websites.

aggregated into botnets that are used to send phishing spam, launch denial-of-service attacks, or distribute malicious code.[34] The harm suffered by the person with the compromised computer is considerably less than the aggregate damage to society and other users. Second, users face greater harm than vendors do, especially overall.[35] While precise figures are difficult to ascertain, reliable estimates of the worldwide economic damage caused by digital attacks in 2003 range from $12.5 billion for worms and viruses, and $226 billion for all attacks,[36] to $157–$192 billion on Windows PCs alone in 2004.[37] Losses to vendors from security breaches, such as from increased support costs, reputational harm, and declines in share price, are also uncertain, but likely considerably smaller.[38] Vendors, therefore, have less incentive to fix bugs than is socially optimal.

The rise of cloud computing and mobile computing worsens the problem. For example, the popular micro-blogging service Twitter suffered a security breach when a hacker cracked an employee's Gmail account, giving him access to business documents stored on Google's Apps service.[39] The hacker then forwarded confidential company documents to the website TechCrunch, which published them.[40] He also took over the e-mail accounts of senior executives and gained access to Twitter employees' phone records, personal e-mail messages, and credit card data.[41] Thus, a weakness in a cloud-computing application—here, the password-recovery feature of Gmail—caused a cascade of harm to multiple users and to their employer.[42] Chinese hackers were able to penetrate Google's security to access accounts of human rights activists by

---

[34] *See, e.g.*, Brett Stone-Gross et al., *Your Botnet Is My Botnet: Analysis of a Botnet Takeover*, 16 PROC. ACM CONF. ON COMPUTER & COMM. SECURITY 635, 635 (2009).

[35] *See* Byung Cho Kim et al., *An Economic Analysis of the Software Market with a Risk-Sharing Mechanism*, INT'L J. ELECTRONIC COM., Winter 2009–2010, at 7, 26–29 (discussing how the security burden falls on the consumer because software vendors are not directly liable for losses incurred due to poor security).

[36] BRIAN CASHELL ET AL., CONG. RESEARCH SERV., RL 32331, THE ECONOMIC IMPACT OF CYBER-ATTACKS 9 tbl.3, 10 tbl.4 (2004).

[37] *$290 of Malware Damage per Windows PC Worldwide in 2004; XP Service Pack 2 Creates "Haves and Have Nots" as Road Forks*, MI2G (Aug. 24, 2004, 5:45 PM), http://www.mi2g.com/cgi/mi2g/frameset.php?pageid=http://www.mi2g.com/cgi/mi2g/press/240804.php.

[38] *See* JOHN VIEGA, THE MYTHS OF SECURITY 142–44 (Mike Loukides ed., 2009).

[39] Claire Cain Miller & Brad Stone, *Twitter Hack Raises Flags on Security*, N.Y. TIMES, July 16, 2009, at B1.

[40] Nik Cubrilovic, *The Anatomy of the Twitter Hack*, TECHCRUNCH (July 19, 2009), http://www.techcrunch.com/2009/07/19/the-anatomy-of-the-twitter-attack.

[41] *Id.*

[42] Miller & Stone, *supra* note 39.

exploiting a security flaw in Microsoft Internet Explorer.[43] Similarly, in March 2009, a flaw in Google Docs briefly exposed private documents to the public, causing the Electronic Privacy Information Center to file a complaint with the Federal Trade Commission charging Google with a deceptive trade practice.[44]

As consumers increasingly store data on, and connect to the internet with, smartphones, vulnerabilities in devices such as Apple's iPhone,[45] and operating systems such as Google's Android,[46] put sensitive personal information at risk. The growing move to use phones for payment systems—whether pay-by-SMS or PayPal—makes hacking phones even more attractive.[47]

Bugs happen. Inevitably, software code is imperfect.[48] While vendors find and fix some flaws, the demands of the release cycle, and the panoply of configurations and interactions that software encounters when deployed by users, ensure that bugs slip through into production code. Some of those bugs create security weaknesses that can be exploited. The research firm Gartner calculates that 75% of security breaches result from software flaws.[49] Even large, security-conscious vendors produce vulnerable code. Oracle faces a new automated tool that allows any minimally skilled computer user to break into the firm's database software over the internet.[50] In October 2009, Microsoft released a record number of fixes for Patch Tuesday—even though its code base did not yet include the new operating system Windows 7.[51] Adobe recently patched a vulnerability in its ubiquitous Acrobat software that allowed

---

[43] *Microsoft Admits Explorer Used in Google China Hack*, BBC NEWS, http://news.bbc.co.uk/2/hi/8460819.stm (last updated Jan. 15, 2010); Riva Richmond, *Microsoft Plugs Security Hole Used in Attacks on Google*, N.Y. TIMES BITS (Jan. 21, 2010, 2:24 PM), http://bits.blogs.nytimes.com/2010/01/21/microsoft-plugs-security-hole-used-in-december-attacks.

[44] Complaint and Request for Injunction, Request for Investigation and for Other Relief, Google, Inc. and Cloud Computing Servs. (F.T.C. Mar. 17, 2009), *available at* http://epic.org/privacy/cloudcomputing/google/ftc031709.pdf.

[45] Elinor Mills, *Researcher Warns of Risks from Rogue iPhone Apps*, CNET INSECURITY COMPLEX (Feb. 3, 2010, 4:00 AM), http://news.cnet.com/8301-27080_3-10446402-245.html (describing "SpyPhone" app).

[46] Dean Takahashi, *Q&A with Charlie Miller on Hacking the T-Mobile G1 Phone with Google Android Software*, VENTUREBEAT (Oct. 28, 2008), http://venturebeat.com/2008/10/28/qa-with-charlie-miller-on-hacking-the-t-mobile-g1-phone-with-google-android-software.

[47] *See* VIEGA, *supra* note 38, at 109–11; Daniel Roth, *The Future of Money: It's Flexible, Frictionless, and (Almost) Free*, WIRED, Mar. 2010, at 70.

[48] *See* VIEGA, *supra* note 38, at 139–44.

[49] STUART OKIN, COMSEC CONSULTING, MANAGING THE COST OF IT SECURITY 4 (2008), http://www.comsecglobal.com/FrameWork/Upload/Managing the cost of IT Security.pdf.

[50] Jim Finkle, *Hacking Oracle's Database Will Soon Get Easier*, REUTERS, July 22, 2009, *available at* http://www.reuters.com/article/2009/07/22/us-oracle-hackers-idUSTRE56L66D20090722.

[51] Dan Goodin, *Microsoft's Patch Tuesday Fixes Record Number of Flaws*, REGISTER (Oct. 14, 2009, 12:09 AM), http://www.theregister.co.uk/2009/10/14/microsoft_patch_tuesday_oct_2009.

hackers to access data on vulnerable computers—nearly a month after the bug was first reported and code to exploit it became available.[52]

Vulnerabilities surface quickly. As Eric Raymond famously observed, "Given enough eyeballs, all bugs are shallow."[53] Some users encounter bugs unexpectedly; others know how to look for them. Hackers are expert in how software fails.[54] While they lack inside information about the software, there are more of them than there are engineers on quality assurance teams even at large software firms, and they are highly motivated—by money, by reputation, and even by ideology. In one week in July 2009, for example, outside researchers released information about security flaws in the Linux operating system kernel,[55] the Mozilla Firefox browser,[56] and the Bluetooth communications protocol.[57] The Chromium open source project has acknowledged that outside researchers found a number of critical bugs in its browser, and Google has begun offering rewards to hackers who find flaws in it.[58]

Moreover, outsiders have the advantage of time. A software company's testers must recheck each new version of a program, and they have a limited period of time to inspect the final code before it is released to the public.[59] Independent researchers can examine the ultimate version at their leisure. In addition, hackers quickly convert information on security flaws into tools for exploiting them. One comprehensive study of vulnerabilities found that over 70% of bugs had exploit code available by the time the flaw was publicly

---

[52] Ryan Naraine, *Adobe Confirms PDF Zero-Day Attacks. Disable JavaScript Now*, ZDNET ZERO DAY (Dec. 15, 2009, 9:08 AM), http://blogs.zdnet.com/security/?p=5119; Ryan Naraine, *Adobe Plugs PDF Zero-Day Flaw in Latest Security Makeover*, ZDNET ZERO DAY (Jan. 13, 2010, 8:06 AM), http://blogs.zdnet.com/security/?p=5234.

[53] RAYMOND, *supra* note 12, at 30 (internal quotation marks omitted).

[54] *See, e.g.*, *Interview: Bruce Schneier*, FRONTLINE, http://www.pbs.org/wgbh/pages/frontline/shows/hackers/interviews/schneier.html (last visited May 13, 2011) ("[Hackers] are the experts in how the systems work and how the systems fail.").

[55] Dennis Fisher, *New Linux Flaw Enables Null Pointer Exploits*, THREATPOST (July 17, 2009, 10:45 AM), http://threatpost.com/en_us/blogs/new-linux-flaw-enables-null-pointer-exploits-071709.

[56] Ryan Paul, *Firefox 3.5.1 Released to Patch TraceMonkey Vulnerability*, ARS TECHNICA (July 19, 2009, 8:35 PM), http://arstechnica.com/open-source/news/2009/07/firefox-351-released-to-patch-tracemonkey-vulnerability.ars.

[57] Sumner Lemon, *HTC Issues Hotfix for Bluetooth Vulnerability in Smartphones*, TECHWORLD (July 20, 2009, 8:29 AM), http://www.techworld.com.au/article/311563/htc_issues_hotfix_bluetooth_vulnerability_smartphones.

[58] Chris Evans, *Encouraging More Chromium Security Research*, CHROMIUM BLOG (Jan. 28, 2010), http://blog.chromium.org/2010/01/encouraging-more-chromium-security.html.

[59] *See* JEFF TIAN, SOFTWARE QUALITY ENGINEERING 4–5 (2005).

disclosed.[60] The incidence of "zero day" bugs—security holes that become public before vendors have software patches ready—is rising sharply.[61] In short, outside researchers will always find flaws that vendors did not catch, and some of those bugs will have serious security repercussions.

This problem for vendors worsens as software becomes more popular and more complex. Operating system (OS) software, for example, is particularly subject to flaws. An OS must expose key aspects of its internal workings to the software development community, creating the possibility that a bug in an application can wreak havoc on the operating system.[62] As more developers write applications for the platform, the OS must maintain backwards compatibility (ensuring that programs written for its earlier versions work with the latest one) and must test an increasing number of software interactions and dependencies. Demands from developers, and the need to ensure that older software continues to run properly, can lead OS vendors to tolerate security flaws that could be eliminated, but at the cost of sacrificing backwards compatibility.

For example, Microsoft maintained a weak, easily cracked password feature (the LAN Manager password hash) in Windows 2000 to avoid breaking numerous third-party applications written for earlier versions of Windows.[63] The trade-off for the Redmond company may have been rational: The benefits of having additional content running on Windows might outweigh the added security risks—to Microsoft—of LAN Manager hacks. However, for versions of the operating system through Windows 2000, LAN Manager was the default authentication method in certain circumstances.[64] Thus, individuals or companies running Windows would be vulnerable unless they actively took steps to prevent an attack.[65]

---

[60] Stefan Frei et al., *Large-Scale Vulnerability Analysis*, 2006 PROC. ACM SIGCOMM WORKSHOP ON LARGE-SCALE ATTACK DEF. 131, 135.

[61] *Id.* ("[T]he number of zero-day exploits is increasing dramatically.").

[62] *See, e.g.*, *Overview of the Windows API*, MICROSOFT DEVELOPER NETWORK, http://msdn.microsoft.com/en-us/library/aa383723(VS.85).aspx (last updated Mar. 25, 2010) ("Windows application programming interface (API) enables [software developers] to exploit the power of Windows. . . . [and] develop applications that run successfully on all versions of Windows . . . .").

[63] CHAD TODD, HACK PROOFING WINDOWS 2000 SERVER 395 (2001).

[64] *Id.* at 394–400.

[65] *Cf.* RICHARD H. THALER & CASS R. SUNSTEIN, NUDGE 85–89 (2008) (discussing effects of default settings).

The LAN Manager example highlights a critical puzzle: vendors do not always act to fix known weaknesses or, at least, to fix them promptly.[66] Indeed, companies may learn about vulnerabilities at no cost, as when independent researchers discover and report bugs.[67] Examples abound. Juniper Networks barred one of its researchers from giving a talk at the Black Hat and Defcon conferences about a vulnerability in Automated Teller Machines (ATMs), even though the affected vendor had known of the flaw for nine months.[68] Apple failed to fix three weaknesses in its iCal scheduling software, despite having four months of advance warning from researchers at Core Security.[69] Sun Microsystems left a vulnerability in its Solaris operating system unpatched for over six months, even though it allowed hackers to crash the software via a buffer overflow exploit.[70]

Bruce Schneier, a security expert, posits two reasons for vendors' lassitude in patching such bugs.[71] First, he notes that harms from vulnerabilities affect vendors far less than customers. Bugs create a negative externality. Furthermore, if customers cannot discern which component of a system is responsible for a problem—for example, detecting whether the flaw was in the operating system, the application, or the data—vendors face reduced reputational or market pressures to improve security.[72] Second, customers tend to value new features or faster releases over slower, more limited, but more secure, software. Added features generate more sales than reduced bugs.

Even if vendors do patch vulnerabilities, they may not call users' attention to the need to install those fixes.[73] Despite vendors' urgings, users cannot

---

[66] *See* Ashish Arora et al., *Optimal Policy for Software Vulnerability Disclosure*, 54 MGMT. SCI. 642 (2008) (arguing that vendors patch later than is socially optimal).

[67] *See, e.g.*, Gregg Keizer, *Microsoft Patches IE, Admits It Knew of Bug Last August*, COMPUTERWORLD (Jan. 21, 2010, 2:52 PM), http://www.computerworld.com/s/article/9147058/Microsoft_patches_IE_admits_it_knew_of_bug_last_August (describing bug reported by Israeli security company).

[68] Robert McMillan, *Juniper Nixes ATM Security Talk*, PCWORLD (June 30, 2009, 3:20 PM), http://www.pcworld.com/businesscenter/article/167648/juniper_nixes_atm_security_talk.html.

[69] John Leyden, *Researchers Out Apple over Unpatched iCal Bugs*, REGISTER (May 22, 2008, 5:05 PM), http://www.theregister.co.uk/2008/05/22/unpatched_apple_bug_flap.

[70] *Security Hole in Sun Solaris Left Unpatched for Months*, THE H (June 9, 2008, 12:46 PM), http://www.h-online.com/newsticker/news/item/Security-hole-in-Sun-Solaris-left-unpatched-for-months-736215.html.

[71] Bruce Schneier, *The Problem Is Information Insecurity*, TECHWATCH TECH BLOG (Aug. 10, 2008), http://www.techwatch.co.uk/2008/08/10/the-problem-is-information-insecurity.

[72] *Cf.* VIEGA, *supra* note 38, at 144 (suggesting that vendors derive a greater benefit from allowing consumers to discover bugs and notify them, rather than from investing their own capital to perfect the system).

[73] *See, e.g.*, Zetter, *supra* note 2.

always install each new patch. Large-scale users, especially corporations, have limited windows in which they can install patches—typically, they do so several times a year to ensure sufficient time to test the stability of those changes in their environments.[74] Thus, even if a vendor releases a patch, customers may not have sufficient information to appreciate the relative necessity of applying it immediately—and those who do may be constrained from patching by the demands of their computing environment. Software companies may also be reluctant to reveal flaws due to fears that disclosure can increase, not decrease, risk.[75] Describing a bug—even in the documentation available with a patch that remedies it—creates hazards.[76] Attackers can use the description to reverse engineer the flaw, and then to create code that exploits it. This highlights the challenge that vendors, and researchers practicing responsible disclosure, face: if they describe flaws with too much precision, hackers can probe the weaknesses, but if they are too general, customers will encounter difficulty taking precautions. Vendors thus prefer to keep vulnerabilities secret, believing this best protects them and their customers while fixes are readied.[77] To improve security, though, software companies need not only to fix vulnerabilities, but to inform users so they can apply those fixes.[78] This is particularly important since black hat hackers—those who employ vulnerability data to compromise systems for gain—frequently have access to information about bugs already.[79]

It may also make economic sense for vendors to underplay bugs. Though users accept that all software has flaws, a vendor may worry about its reputation relative to competitors if it publicizes widely each bug it patches.[80] The concern is strategic behavior: a competing vendor who keeps vulnerabilities quiet may enjoy an advantage in perception, even if its underlying code is no more secure. Absent information to detect this strategy,

---

[74] *See Update Management Process*, MICROSOFT TECHNET, http://technet.microsoft.com/en-us/library/cc700845.aspx (last updated June 1, 2007).

[75] Cross, *supra* note 25, at 39–40.

[76] *See, e.g.*, CHRISTIAN A. CHRISTIANSEN, INT'L DATA CORP., WHAT ENTERPRISES SHOULD KNOW ABOUT VENDOR BEST PRACTICES FOR SECURITY PATH ISSUANCE 3 (2008) (on file with authors) (advocating "providing just enough information on a vulnerability to help mitigate risk, but not so much information that a patch can be reverse engineered").

[77] *Id.* at 2–3 (describing "silent fixing").

[78] *See* Bruce Schneier, *The Nonsecurity of Secrecy*, COMM. ACM, Oct. 2004, at 120, 120 ("We are all less secure if software vendors don't make their security vulnerabilities public . . . .").

[79] CHRISTIANSEN, *supra* note 76, at 2–3.

[80] *See* ORIGINAL SOFTWARE, SOFTWARE QUALITY AND TESTING: A CIO PERSPECTIVE (2008), *available at* http://www.origsoft.com/_assets/client/docs/pdf/whitepapers/cio_software_testing_survey.pdf.

users may assume that the number and severity of reported (and even patched) bugs correlates with software quality.[81] Rational vendors would thus tend to report bugs less frequently, and with less dissemination, than would be socially optimal.

The other key aspect of vendors' decisions is that fixing vulnerabilities is costly. The cost to fix a bug increases as the software development cycle progresses; once the code is in production and use—when independent researchers typically first get access to it—the cost is greatest.[82] One study of United Kingdom businesses found that for every dollar spent on software development, a company spent seventy-five cents on average to remediate security flaws.[83] Analysts agree that fixing security bugs is expensive, though quantifying those costs with precision is difficult. Vulnerabilities in web applications may cost as little as $400 per flaw to fix, while a cross-platform vulnerability in software such as Oracle's can require over $1 million.[84] An IDC study found that fixing bugs in applications developed in-house by corporations costs from $5 million to $22 million per year, depending on the organization's size.[85]

Software code inevitably has weaknesses that internal testing fails to discover. Outsiders find these flaws in time. Even if they report bugs to the vendor, that company may not fix the problems for financial or reputational reasons. This can generate conflicts with outside researchers, whose behavior and motivations are explored in the next section.

B. Bug Hunters

Independent security researchers are a puzzle: they find bugs for free, even when software firms normally pay for this work. Broadly speaking, there are three types of testers: software company employees, consultants, and independent researchers. Employees—generally called Quality Assurance or

---

[81] *See* CHRISTIANSEN, *supra* note 76, at 2.

[82] Vance J. VanDoren, *How Communications Help Integration Projects Succeed*, CONTROL ENGINEERING, Apr. 1, 2009, at 42, 43 fig.

[83] Warwick Ashford, *On-Demand Service Aims to Cut Cost of Fixing Software Security Flaws*, COMPUTERWEEKLY.COM (July 14, 2009, 1:00 AM), http://www.computerweekly.com/Articles/2009/07/14/236875/on-demand-service-aims-to-cut-cost-of-fixing-software-security.htm.

[84] Kelly Jackson Higgins, *The Cost of Fixing an Application Vulnerability*, DARK READING (May 11, 2009, 1:56 PM), http://www.darkreading.com/security/app-security/showArticle.jhtml?articleID=217400256 (analyzing results from United Kingdom firms).

[85] Joy Persaud, *Cost of Fixing Software Defects 'Runs into Millions,'* SC MAG. (July 18, 2008), http://www.scmagazineuk.com/Cost-of-fixing-software-defects-runs-into-millions/article/112597.

Quality Engineering—are compensated directly for their work by the software vendor (their employer).[86] Consultants, too, earn remuneration from the vendor by searching for flaws under contract. Independent researchers, though, are neither paid by nor affiliated with the vendor. They might benefit indirectly from their tasks, such as when a tester uses the software herself, works for a firm that does so, or employs testing as a signal of skill or experience.[87] But most hackers have an attenuated relationship at most with the producer of the code they try to break.

Independent researchers test for a variety of reasons: possible future remuneration, intellectual satisfaction, peer recognition, ideological commitment, animus toward a particular vendor, and expectations in a larger community of testers, among others.[88] In short, their incentives are diverse. Their actions represent another example of peer production—creation outside the standard market economy through a disaggregated, informal process. Scholars such as Jonathan Zittrain,[89] Yochai Benkler,[90] Eric S. Raymond,[91] and Eric von Hippel[92] have analyzed other instances of peer production, from mapping craters on Mars,[93] to Wikipedia,[94] to open source software,[95] to kitesurfing.[96] Independent bug hunters analyze software for many reasons, but few are linked directly to financial incentives.

Thus, the standard incentives-based model for intellectual property does not apply to hackers. Researchers who find bugs can rarely obtain IP protection for their discoveries, if at all.[97] Vulnerability data consists of insight into how code functions, or fails to do so, and would be unprotectable under copyright as either an idea, or as an unauthorized derivative work of the underlying

---

86 TIAN, *supra* note 59.

87 *See* VON HIPPEL, THE SOURCES OF INNOVATION, *supra* note 11.

88 *See generally* BRUCE STERLING, THE HACKER CRACKDOWN (1992) (reviewing the development of the hacker subculture in cyberspace).

89 JONATHAN ZITTRAIN, THE FUTURE OF THE INTERNET—AND HOW TO STOP IT (2008).

90 Benkler, *supra* note 10.

91 RAYMOND, *supra* note 12.

92 *See* sources cited *supra* note 11.

93 *See Dawn Mission: Clickworkers*, NASA, http://dawn.jpl.nasa.gov/clickworkers (last visited May 13, 2011).

94 *See* YOCHAI BENKLER, THE WEALTH OF NETWORKS 70–74 (2006).

95 *Id.* at 63–67.

96 VON HIPPEL, DEMOCRATIZING INNOVATION, *supra* note 11, at 103–04.

97 *See, e.g.*, 17 U.S.C. § 103(a) (2006) (denying copyright protection to unauthorized derivative works); Derek E. Bambauer, *Faulty Math: The Economics of Legalizing* The Grey Album, 59 ALA. L. REV. 345, 348–54 (2008) (describing how the adaptation right impedes creation of valuable derivative works).

program.[98] While a patent on the bug might theoretically be available, the time lag for prosecution and the existence of the program as prior art make this possibility largely irrelevant.[99] The incentives generated by IP law do little to spur independent researchers to test code.

Intellectual property doctrine does have a more subtle, second-order effect on researchers' behavior, but it affects how they distribute vulnerability data rather than whether they produce it. Vendors' goals for distribution effects are straightforward: they want to be the sole recipients of bug data. IP law can be deployed to shape when, and with whom, hackers share information regardless of any effects on whether they conduct such research to begin with. Put crudely, vendors frequently employ IP law as a gag order.

This approach is significantly misguided. Researchers have an easy and profitable distribution alternative to sharing data with vendors: they can sell their discoveries on the black market.[100] Organized crime entities, malware operators, and governments pay well for vulnerabilities in important software products, particularly those with no known patch or defense. Selling bug data to the underground is appealing for several reasons. First, it is hard for vendors to detect these transactions: research on bugs takes place in private, as does the exchange of exploits for money. Second, the black market typically pays better for bugs than the legitimate market does.[101] In 2006, antivirus vendor Trend Micro found that code exploiting a vulnerability sells for $20,000–$30,000 depending upon how widely used the insecure software is and how reliable the exploit code is, while code that takes over a PC and adds it to a botnet goes for roughly $5,000.[102] Finally, the legal risks are, ironically, likely lower in this setting. If what a hacker does to enumerate a flaw is

---

[98] 17 U.S.C. § 102(b); *cf.* Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807 (1st Cir. 1995) (holding Lotus 1-2-3 menu command structure ineligible for copyright protection), *aff'd by an equally divided court*, 516 U.S. 233 (1996).

[99] In 2010, the average time from filing to patent issuance or abandonment was 35.3 months. U.S. PATENT & TRADEMARK OFFICE, PERFORMANCE AND ACCOUNTABILITY REPORT FISCAL YEAR 2010, at 19 tbl. (2010), *available at* http://www.uspto.gov/about/stratplan/ar/2010/USPTOFY2010PAR.pdf.

[100] Jaziar Radianti & Jose. J. Gonzalez, *Toward a Dynamic Modeling of the Vulnerability Black Market* 4–7 (Workshop on the Econ. of Securing the Info. Infrastructure, No. 4898, 2006), *available at* http://wesii.econinfosec.org/draft.php?paper_id=44.

[101] *See* Andy Greenberg, *A Hacker's Nasdaq*, FORBES.COM (Aug. 9, 2007, 6:00 AM), http://www.forbes.com/2007/07/06/security-software-hacking-tech-security-cx_ag_0706vulnmarket.html ("'It's hard to say no if the black market offers you $300,000,' Aitel [chief technology officer of vulnerabilities broker, Immunity] says.").

[102] Ryan Naraine, *Hackers Selling Vista Zero-Day Exploit*, EWEEK.COM (Dec. 15, 2006), http://www.eweek.com/c/a/Security/Hackers-Selling-Vista-ZeroDay-Exploit.

potentially unlawful, she runs less risk by concealing this activity than by announcing it to the vendor, the security community, or the larger public. Both participants to the illegal transaction have incentives to conceal it, and while they may face criminal penalties if caught, their risk of detection is typically low.[103] The risk is clear: if hackers fear lawsuits for publishing vulnerability discoveries, they can opt to sell their findings on the black market at lower risk and greater reward, placing users at risk.

Thus, intellectual property has but a muted effect on the production of vulnerability data by independent security researchers, but can have a profound effect on its distribution. However, this impact is perverse: rather than push bug hunters into sharing information with vendors, it increases the attractiveness of distribution through illicit channels to consumers who are likely bad actors. The black market is discreet and profitable. Nonetheless, vendors continue to deploy a range of IP-based legal weapons in an attempt to control researchers. The next Part examines these tools.

II. THE VENDOR'S ARSENAL

A. Copyright: Breaking the Censor's Scissors

The Great Firewall of China has holes. Scholars at the University of Michigan found key flaws in part of the firewall, Green Dam-Youth Escort, created by Jinhui Computer System Engineering (JCSE or Jinhui). JCSE built Green Dam to augment China's formidable internet censorship apparatus; the Chinese government mandated that all computer manufacturers install—or at least ship—the firewall software on every new computer.[104] Green Dam not only censored users' internet access, it created significant security risks. The Michigan researchers found that vulnerabilities in the code could permit malicious websites to take control of a user's computer to steal personal information or to enlist the PC in a botnet.[105]

---

[103] *See, e.g.*, 18 U.S.C. § 1030 (2006) (amended 2008) (penalizing unauthorized access to computer systems with imprisonment and fines).

[104] Edward Wong & Ashlee Vance, *China Intent on Requiring Internet Censor Software*, N.Y. TIMES, June 19, 2009, at A10.

[105] Scott Wolchok et al., *Analysis of the Green Dam Censorware System*, COMPUTER SCI. & ENG'G AT THE UNIV. OF MICH. (June 18, 2009), http://www.cse.umich.edu/~jhalderm/pub/gd.

Jinhui—already under public pressure for helping China's government censor the internet—responded with indignation and a threat.[106] The company's general manager stated, "It is not responsible to crack somebody's software and publish the details, which are commercial secrets, on the Internet. [The Michigan researchers] have infringed the copyright of our product."[107] He added that Jinhui planned to sue the researchers.[108] While a suit in the United States would likely fail—reverse engineering is protected as fair use under U.S. copyright law[109] and is considered a legitimate practice by trade secret doctrine[110]—and the Michigan team would have little to fear from legal action in China, this response typifies vendors' reactions to public disclosure of vulnerabilities. Jinhui patched some of the flaws that the Michigan team found, yet simultaneously threatened them for performing free quality control.[111] To software companies like JSCE, the perception of security is often more important than security itself. And when that perception is threatened, intellectual property threats are often their first response.

The following sections detail the IP theories that software companies use, and the doctrinal adjustments this Article argues are necessary to protect security research.

B. Patent

Chris Paget was going to give a presentation at the Black Hat conference in 2007 that would show how to clone an RFID (radio frequency identifier) chip—the kind used in cards to control access to buildings, in tags that allow drivers to pay tolls without stopping, and in passports to verify one's identity.[112] His subject was an RFID sensor made by HID Global; Paget chose the company because it produced the ID cards in the building where his employer, IOActive, was located.[113]

---

[106] Edward Wong, *China: Artist Urges Online Boycott*, N.Y. TIMES, June 23, 2009, at A11; *Green Dam Breached, Patch-Up in Progress*, PEOPLE'S DAILY ONLINE (June 15, 2009, 8:39 AM), http://english.people.com.cn/90001/90776/90882/6678151.html.

[107] *Green Dam Breached, Patch-Up in Progress*, *supra* note 106.

[108] *Id.*

[109] *See, e.g.*, Sony Computer Entm't, Inc. v. Connectix Corp., 203 F.3d 596 (9th Cir. 2000).

[110] *See, e.g.*, Chi. Lock Co. v. Fanberg, 676 F.2d 400, 404 (9th Cir. 1982).

[111] *Green Dam Breached, Patch-Up in Progress*, *supra* note 106.

[112] Paul F. Roberts, *Lawsuits, Patent Claims Silence Black Hat Talk*, INFOWORLD (Feb. 27, 2007, 9:30 AM), http://www.infoworld.com/d/security-central/lawsuits-patent-claims-silence-black-hat-talk-720.

[113] Paul F. Roberts, *Battle Brewing over RFID Chip-Hacking Demo*, INFOWORLD (Feb. 26, 2007, 3:45 PM), http://www.infoworld.com/article/07/02/26/HNblackhatrfid_1.html.

HID Global objected, strongly. Their letter to IOActive demanded that the research firm not publish information about how to "spoof" HID's cards, or face legal action for patent infringement.[114] HID asserted Paget's cloning would violate two of its patents, which cover an identification system using passive integrated transponders.[115] The threat created significant legal risk for Paget and IOActive: if their cloner was covered by the claims of one or more of HID's patents and they proceeded in the face of the vendor's warnings, they would be liable for willful patent infringement. Willful infringement subjects a defendant to increased damages—up to three times actual damages[116]—and attorney's fees.[117] Patent law operates under strict liability: if Paget's actions constituted using HID's invention, he would be liable, regardless of how laudatory his purpose. He and IOActive decided not to give the offending presentation, and Black Hat staffers tore their prepared materials out of the conference packets.[118] Instead, the researchers gave a generic presentation, with no mention of HID or its technology.[119]

Patent protection is among the most powerful weapons a software vendor can deploy to control its code. Patent law confers a monopoly over making, using, selling, or offering to sell the protected invention.[120] Infringement operates under strict liability: anyone who creates a product, or performs a process, that incorporates all elements listed in a patent's claims violates the patent owner's rights.[121] Defenses are scant,[122] and damages are at least a

---

114 Letter from HID Global to IOActive (Feb. 21, 2007) (on file with authors).

115 *See* U.S. Patent No. 5,041,826 (filed Feb. 16, 1990); U.S. Patent No. 5,166,676 (filed Feb. 16, 1990).

116 35 U.S.C. § 284 (2006); *see also In re* Seagate Tech., LLC, 497 F.3d 1360, 1368 (Fed. Cir. 2007) (noting that the court earlier held that "an award of enhanced damages requires a showing of willful infringement").

117 35 U.S.C. § 285; *see also* Mahurkar v. C.R. Bard, Inc., 79 F.3d 1572, 1579 (Fed. Cir. 1996) (attorney's fees available in exceptional cases).

118 Ryan Naraine, *Legal Threat Forces Cancellation of Black Hat RFID Hacking Demo*, ZDNET ZERO DAY (Feb. 27, 2007, 6:50 AM), http://blogs.zdnet.com/security/?p=102&tag=col1;post-103.

119 Larry Greenemeier, *Black Hat 'RFID' Compromise Is a Win for Security*, INFORMATIONWEEK (Feb. 28, 2007), http://www.informationweek.com/news/internet/showArticle.jhtml?articleID=197700210.

120 35 U.S.C. § 271(a).

121 *See, e.g.*, Bio-Tech. Gen. Corp. v. Genentech, Inc., 80 F.3d 1553, 1559 (Fed. Cir. 1996).

122 35 U.S.C. § 271(e) (amended 2010) (permitting limited infringement for regulatory data submission); *id.* § 273 (defining prior inventor defense); *id.* § 282 (defining noninfringement, unenforceability, invalidity, and failure to comply with reissue requirements); C.R. Bard, Inc. v. M3 Sys., Inc., 157 F.3d 1340, 1372–73 (Fed. Cir. 1998) (defining patent misuse). *But see* 35 U.S.C. § 271(d) (listing conduct not qualifying as misuse); Akron Polymer Container, Corp. v. Exxel Container, Inc., 148 F.3d 1380, 1383 (Fed. Cir. 1998) (defining inequitable conduct before the U.S. Patent and Trademark Office); Intel Corp. v. ULSI Sys. Tech., Inc., 995 F.2d 1566, 1568 (Fed. Cir. 1993) (defining first use or patent exhaustion); Met-Coil Sys. Corp. v. Korners Unlimited, Inc., 803 F.2d 684, 686 (Fed. Cir. 1986) (defining implied license).

reasonable royalty for use plus interest.[123] There is no fair use in patent law: at most, an "experimental use" exception immunizes use of an invention for "amusement, to satisfy idle curiosity, or for strictly philosophical inquiry."[124] Any commercial use places an infringer outside this safe harbor.[125] Thus, patent law incorporates none of the utility calculus present in the copyright[126] or trademark[127] fair use defenses; no matter how beneficial a researcher's findings, if they are obtained in violation of a patent, without authorization, the researcher is liable.

Paget and IOActive made the rational choice to alter their presentation. But society is ill served by patent's strict liability in the software security context. RFID tags, for example, are becoming ubiquitous, appearing in subway fare cards,[128] animal identification implants,[129] library books,[130] bicycle race trackers,[131] and shipments of Oxycontin.[132] Vulnerabilities in their operation[133] have become particularly worrisome now that RFIDs play a key role in governmental operations such as border control (passports[134]) and Department of Defense procurement.[135] Paget demonstrated the risks: with a few hundred dollars of equipment loaded into his Volvo, he was able to "skim" the serial numbers for six passport cards within an hour of driving along San

---

123 35 U.S.C. § 284.

124 Roche Prods., Inc. v. Bolar Pharm. Co., 733 F.2d 858, 863 (Fed. Cir. 1984), *superseded on other grounds by statute*, Drug Price Competition and Patent Term Restoration Act of 1984, Pub. L. No. 98-417, § 202, 98 Stat. 1585, 1603 (codified at 35 U.S.C. § 271(e)(1) (2006)) (permitting acts otherwise considered infringement, such as in *Bolar*, when in the narrow context of pursuing FDA approval).

125 Madey v. Duke Univ., 307 F.3d 1351, 1362 (Fed. Cir. 2002).

126 *See, e.g.*, Campbell v. Acuff-Rose Music, Inc., 510 U.S. 569, 577–78 (1994).

127 *See, e.g.*, New Kids on the Block v. News Am. Publ'g, Inc., 971 F.2d 302, 306–08 (9th Cir. 1992) (describing fair use).

128 Andrew Heining, *Another RFID Smart Card Vulnerability Exposed*, CHRISTIAN SCI. MONITOR (Oct. 8, 2008, 1:10 PM), http://features.csmonitor.com/innovation/2008/10/08/another-rfid-smart-card-vulnerability-exposed.

129 *See, e.g.*, U.S. DEP'T OF AGRIC., THE NATIONAL ANIMAL IDENTIFICATION SYSTEM (NAIS) (2007), *available at* http://www.ftcldf.org/docs/R.pdf.

130 *See, e.g.*, David Molnar & David Wagner, *Privacy and Security in Library RFID: Issues, Practices, and Architectures*, 11 PROC. ACM CONF. ON COMPUTER & COMM. SECURITY 210 (2004).

131 Daniel Lee, *Just 1 Minute*, INDIANAPOLIS STAR, Apr. 25, 2006, at 1C.

132 Elena Malykhina, *Drugmaker Ships RFID Tags with OxyContin*, INFORMATIONWEEK, Nov. 22, 2004, at 20.

133 *See, e.g.*, Russell Ryan et al., Anatomy of a Subway Hack (2008) (PowerPoint presentation), http://tech.mit.edu/V128/N30/subway/Defcon_Presentation.pdf (describing how MIT students hacked the RFID-based Charlie Card system for the Boston subway system).

134 *See, e.g.*, *U.S. Passport Card*, TRAVEL.STATE.GOV, http://travel.state.gov/passport/ppt_card/ppt_card_3926.html (last visited May 13, 2011).

135 *See, e.g.*, Radio Frequency Identification, 48 C.F.R. § 252.211–7006 (2010).

Francisco's Fisherman's Wharf.[136] Despite these problems, researchers cannot lawfully test HID's RFID chips without authorization, and even if they go ahead without permission, they cannot legally distribute their findings, since doing so would prove infringement. HID thus obtains an effective veto over probing its patented products to assess their security.

Change to protect software security research could come from revising the Patent Act or reinterpreting the experimental use exception.[137] Legislation to exempt security research from infringement would be in line with prior moves to create exceptions to liability. Congress has previously created exemptions for socially beneficial uses that would otherwise infringe, from protecting doctors against liability for using protected surgical methods[138] to allowing prior users of a patented business method to continue employing it.[139] Creating a narrow exception to patent liability for security research would equally shield helpful activity that would otherwise be subject to injunctive prohibition.[140] Unlike the surgical methods exemption, though, any protection for security research would likely need to protect tools specifically adapted to the patented method, since software programs needed to probe a vulnerability could otherwise infringe contributorily.[141]

Alternatively, federal courts (in particular the Federal Circuit) could reinterpret the experimental use exception to cover security research.[142] This would necessitate extending immunity to commercial uses of a patent; current doctrine mandates that a defendant's activity be noncommercial.[143] Security research is often commercial, even if indirectly, and thus even widening the ambit of the common law exemption might not ameliorate the chilling effects.

---

136 *Chips in Official IDs Raise Privacy Fears*, FOXNEWS.COM, July 11, 2009, http://www.foxnews.com/story/0,2933,531720,00.html.

137 The much-anticipated Supreme Court decision in *Bilski v. Kappos*, 130 S. Ct. 3218 (2010), addressing the eligibility of business methods for patenting under 35 U.S.C. § 101 (2006), does not appear to have changed the situation facing researchers. Software remains eligible subject matter for patent protection, assuming it meets the other requirements for patentability. *See Bilski*, 130 S. Ct. at 3227–28.

138 35 U.S.C. § 287(c).

139 *Id.* § 273(b).

140 Injunctive relief for patent infringement is typical, but not automatic. eBay Inc. v. MercExchange, L.L.C., 547 U.S. 388, 391 (2006).

141 *See* 35 U.S.C. § 287(c)(2)(A) (excluding from the exemption "use of a patented machine, manufacture, or composition of matter in violation of [a surgical method] patent").

142 *See* Deuterium Corp. v. United States, 19 Cl. Ct. 624, 632 & n.14 (1990) (noting that 35 U.S.C. § 271(e)(1) did not alter the *Bolar* definition of the exception, but rather "changed the narrow application of the doctrine affecting reporting requirements for federal drug laws").

143 Madey v. Duke Univ., 307 F.3d 1351, 1362 (Fed. Cir. 2002). *See generally* 5 DONALD S. CHISUM, CHISUM ON PATENTS § 16.03[1] (2010) (collecting experimental use cases).

The current experimental use exception tracks a bright-line divide between commercial and philosophical activity, rather than weighing the costs and benefits of the infringing acts. However, given the history of specialized legislative exceptions in patent law, courts are probably more likely to defer to Congress than to engage in particularized cost–benefit analysis.[144]

Patent law is a potent weapon for vendors who seek to limit creation and dissemination of vulnerability data. While reinterpreting the experimental use exception to liability could help researchers, it is more likely that congressional action to establish a security research exception is necessary to overcome patent law's negative effects.

C. Trade Secret

There may be such a thing as a free lunch—and free laundry, and soft drinks—at schools that use the Blackboard Transaction System (BTS).[145] BTS lets students use their identification cards to pay for goods and services on campus. Billy Hoffman, a student at the Georgia Institute of Technology, and Virgil Griffith, a student at the University of Alabama at Tuscaloosa, discovered fundamental flaws in Blackboard's system.[146] For example, BTS did not encrypt the data involved in processing a purchase; instead, the system depended on physical security to prevent access to the data.[147] However, Hoffman easily bypassed Georgia Tech's physical restrictions with a low-tech hack: he removed four screws holding a locked machine door in place with a "cheap metal knife."[148] Doing so gave him access to the devices that controlled the laundry room in which the box was located—and potentially to the rest of the system as well.[149] Hoffman could now perform a replay attack

---

[144] *See* 35 U.S.C. § 271(e)(1) (granting exemption from infringement liability for uses related to developing and submitting data under federal law regulating drugs).

[145] *See* BLACKBOARD, BLACKBOARD TRANSACTION SYSTEM (2004), *available at* http://library.blackboard.com/docs/CS/Bb_Transaction_System_Brochure.pdf.

[146] Virginia Heffernan, *Internet Man of Mystery*, N.Y. TIMES, Nov. 23, 2008 (Magazine), at 38; Michael Margolis, *Card Systems Prove Insecure*, W. COURIER (Macomb, Ill.), Apr. 25, 2003, http://media.www.westerncourier.com/media/storage/paper650/news/2003/04/25/News/Card-Systems.Prove.Insecure-445193.shtml.

[147] John R. Hall, *Blackboard Transaction System Cease and Desist FAQ*, YAK'S LAIR, http://www.yak.net/mirrors/bb-faq.html (last visited May 13, 2011).

[148] Farhad Manjoo, *The Copyright Cops Strike Again*, SALON.COM (Apr. 15, 2003), http://dir.salon.com/story/tech/feature/2003/04/15/acidus.

[149] Acidus, *CampusWide Wide Open*, CRYPTOME, http://cryptome.org/campuswide.txt (last visited May 13, 2011). Hoffman published under the pseudonym Acidus. Margolis, *supra* note 146.

against BTS.[150] By monitoring communication over the BTS network during a transaction, Hoffman could duplicate it, giving him an unlimited supply of free laundry cycles and beverages.

Hoffman contacted Blackboard about his findings, but claimed he was "blown off" by the company.[151] He and Griffith subsequently planned to present their research on BTS vulnerabilities at the Interz0ne computer conference in Atlanta, Georgia.[152] In addition, Hoffman wrote an article covering BTS weaknesses under a pseudonym for the hacker magazine *2600: The Hacker Quarterly*. He closed the piece by stating, "Hopefully this article will force Blackboard to change to a more secure system."[153]

It didn't. But Blackboard did manage to change Hoffman's proposed talk. The day before Hoffman and Griffith were scheduled to present at Interz0ne, Blackboard obtained a temporary injunction from a Georgia state court blocking them from: discussing signal traffic on a BTS network; revealing how information was stored in the BTS system or readers; describing how to create compatible readers; releasing Blackboard emulation code or hardware; or claiming they could provide products or services that legitimately could interact with a Blackboard product.[154] The injunction also required the students to remove any such information from their websites.[155] Finally, Blackboard sent a letter to Interz0ne's conference chair stating that the conference could be held liable, even criminally liable, if it permitted Hoffman and Griffith to present their research, or if Interz0ne failed to remove information about BTS from its materials.[156]

Blackboard relied on several legal theories to bolster its case for the temporary injunction, including violations of the Federal Electronic Communications Privacy Act and Computer Fraud and Abuse Act (CFAA),[157]

---

150 Acidus, *supra* note 149.

151 Manjoo, *supra* note 148.

152 Margolis, *supra* note 146.

153 Acidus, *supra* note 149.

154 Order Temporarily Enjoining Billy Hoffman and Virgil Griffith, Blackboard Inc. v. Hoffman, No. 1:03-CV-1279 (CC) (Ga. Super. Ct. Apr. 12, 2003) [hereinafter Order], *available at* http://www.eff.org/files/filenode/Blackboard_v_Hoffman/20020412-Blackboard-TRO.pdf.

155 *Id.* at 2.

156 Cease and Desist Letter from Gregory S. Smith, Attorney for Blackboard Inc., to Interz0ne II Conference Chair (Apr. 11, 2003), *available at* http://www.interz0ne.com/events/interz0ne_cease_order.html.

157 The complaint refers to the "Consumer Fraud and Abuse Act" but cites the statutory sections for the *Computer* Fraud and Abuse Act "18 U.S.C. § 1030 *et seq*." Verified Complaint at 7, Blackboard Inc. v.

the Georgia Computer Systems Protection Act, and the Lanham Act, but focused principally on the Georgia Trade Secrets Act.[158] Its complaint repeatedly referred to Hoffman's hacker background, attempting to balance dire descriptions of the threat from his presentation with qualifications about the accuracy of Hoffman's claims about BTS security.[159] Blackboard claimed that the presentation risked "massive fraud, security breaches, and other harms, threatening both the physical and financial security of college students."[160]

Tellingly, though, Blackboard stated publicly that it wasn't "really worried about [the] *security* of the system," but instead was "worried about the *reputation* of the system."[161] Hoffman and Griffith focused attention on BTS's dependence on physical protection for the system's security—and on how readily those physical methods could be compromised. Blackboard compared the students' research to breaking into an ATM, eliding the far greater security protections built into those machines.[162] The company sought both to minimize the findings, calling Hoffman a mere "vandal," while also justifying the ban on disclosing information about the vulnerability with warnings about the risk Hoffman created.[163] The restraining order, and a subsequent confidential settlement, blocked Hoffman and Griffith from presenting at Interz0ne, though the ensuing publicity drew attention to the BTS flaws.[164] Trade secret triumphed over toolboxes.

Trade secret law protects information that has economic value because it is not generally known, and that is the subject of efforts to maintain its secrecy.[165] Examples include customer lists, Google's algorithm for building search results,[166] and the formula for Coca-Cola.[167] Software can qualify for trade

---

Hoffman, No. 1:03-CV-1279 (CC) (Ga. Super. Ct. Apr. 11, 2003), *available at* http://www.eff.org/files/filenode/Blackboard_v_Hoffman/20020411-Blackboard-complaint.pdf.

158 *Id.* at 5–8.

159 *Id.* at 2–4, 8.

160 *Id.* at 8.

161 Anitha Reddy, *Blackboard Gets Gag Order Against Smart-Card Hackers*, WASH. POST, Apr. 18, 2003, at E1 (emphases added).

162 *See, e.g.*, Andrea L. Foster, *Judge Prevents Students from Discussing Security of Debit-Card System*, CHRON. HIGHER EDUC. (D.C.), May 2, 2003, at 40.

163 *Blackboard Statement on Client System Security*, BLACKBOARD, http://library.blackboard.com/docs/Statement_on_System_Security.pdf (last visited May 13, 2011); *see also* Hall, *supra* note 147.

164 Heffernan, *supra* note 146.

165 *See, e.g.*, UNIF. TRADE SECRETS ACT (amended 1985), 14 U.L.A. 529 (2005).

166 Tom McNichol, *Can Microsoft's Bing Take a Bite out of Google?*, TIME (July 31, 2009), http://www.time.com/time/business/article/0,8599,1913841,00.html.

167 *2 Sentenced in Coke Trade Secret Case*, CNNMONEY.COM (May 23, 2007, 3:54 PM), http://money.cnn.com/2007/05/23/news/newsmakers/coke/index.htm.

secret status, even when the compiled object code is sold to the public.[168] Trade secrets are protected by injunctions preventing their disclosure[169] when a defendant has obtained them through improper means;[170] holders can also obtain damages.[171] However, trade secret laws expressly permit the use of reverse engineering to discover protected information; only the acquisition of a secret through improper means creates liability.[172] Where researchers obtain information about security flaws in software or hardware through reverse engineering, their subsequent use and disclosure of that information is beyond the reach of trade secret liability.[173] Accordingly, reform to protect researchers may require only that judges scrutinize trade secret claims more searchingly when reverse engineering is involved.

However, there are at least two complications with trade secret law and software research. First, software vendors often include language in the end-user license agreement governing their software that forbids reverse engineering.[174] While such language would be unlikely to create liability for copyright infringement, since limited reverse engineering typically qualifies as fair use,[175] the contractual obligation might be sufficient to make a software user responsible for maintaining the trade secret.[176] Copyright law faces a similar question when end-user license agreements prohibit reverse

---

168 *See, e.g.*, Rivendell Forest Prods. v. Georgia-Pacific Corp., 28 F.3d 1042 (10th Cir. 1994).

169 *See, e.g.*, Lamb-Weston, Inc. v. McCain Foods, Ltd., 941 F.2d 970 (9th Cir. 1991).

170 *See, e.g.*, CAL. CIV. CODE § 3426.1(a) (West 2010) (defining "[i]mproper means" (internal quotation marks omitted)).

171 *See, e.g.*, Sikes v. McGraw-Edison Co., 665 F.2d 731, 736–37 (5th Cir. 1982).

172 Chi. Lock Co. v. Fanberg, 676 F.2d 400 (9th Cir. 1982); Data Gen. Corp. v. Digital Computer Controls, Inc., 357 A.2d 105 (Del. Ch. 1975). *See generally* RESTATEMENT (FIRST) OF TORTS § 757 (1939).

173 The Georgia Trade Secrets Act expressly exempts from liability information acquired from reverse engineering. GA. CODE ANN. § 10-1-761(1) (2009) ("Reverse engineering of a trade secret not acquired by misappropriation or independent development shall not be considered improper means.").

174 For example, Microsoft expressly forbids reverse engineering in the license agreement for its Windows XP operating system. MICROSOFT, MICROSOFT WINDOWS XP HOME EDITION (RETAIL) END-USER LICENSE AGREEMENT FOR MICROSOFT SOFTWARE § 4 (2004), *available at* http://www.microsoft.com/windowsxp/eula/home.mspx. Some vendors are even more restrictive: Network Associates promulgated an end-user license agreement that required the company's approval before publishing reviews or disclosing results from testing the software. Ed Foster, *Some New Shrink-Wrap License Terms Seem Tailor-Made for UCITA*, INFOWORLD, Mar. 5, 2001, at 82, 87. The New York state attorney general filed suit to block this provision as contrary to consumer protection laws and won an order prohibiting Network Associates from enforcing it. People v. Network Assocs., Inc., 758 N.Y.S.2d 466, 471 (Sup. Ct. 2003) (granting injunction requested by attorney general).

175 *See, e.g.*, Sony Computer Entm't, Inc. v. Connectix Corp., 203 F.3d 596, 603–04 (9th Cir. 2000).

176 *See, e.g.*, Trandes Corp. v. Guy F. Atkinson Co., 996 F.2d 655, 664 (4th Cir. 1993).

engineering; even where decompilation would be protected by fair use,[177] such action would create liability for breach of contract.[178] The effort to regulate contract law for computer information transactions via the Uniform Computer Information Transactions Act (UCITA), for example, recognized the potential adverse effects of such contractual provisions.[179] Indeed, the 2002 version of UCITA prohibited contracts banning reverse engineering by declaring them unenforceable.[180] However, UCITA's effects have been minimal, as only two states have transposed its provisions into law.[181] Thus, vendors may be able to circumvent trade secret's safe harbor via contract.

Second, some researchers may switch sides, working first as an employee or consultant, and then moving to perform independent testing. In this case, the software company may have a plausible claim that the researcher's work is influenced by her knowledge of the firm's trade secrets. Mike Lynn's situation with Cisco exemplifies this problem; Lynn began his work on the flaws in Cisco's routers while covered by a nondisclosure agreement.[182] Once he resigned from his position with ISS, Lynn transitioned to independent research, but Cisco argued, plausibly, that his work was influenced by exposure to Cisco's proprietary information.[183]

The case for legal reform in the trade secret context is more compelling for reverse engineering than for researchers who switch sides. Allowing software companies to reify a license agreement into a trade secret claim would confer complete control over research into vulnerabilities in software that involves any decompilation or reverse engineering—which most security testing does.[184] Accepting the terms of a software end-user license agreement is

---

[177] *See, e.g.*, Atari Games Corp. v. Nintendo of Am. Inc., 975 F.2d 832, 843 (Fed. Cir. 1992) (holding reverse engineering can constitute fair use).

[178] *See, e.g.*, Bowers v. Baystate Techs., Inc., 320 F.3d 1317, 1323–28 (Fed. Cir. 2003) (holding restrictions consistent with the Copyright Act, finding breach of a license agreement prohibiting reverse engineering, and affirming monetary damages award).

[179] We thank Maureen O'Rourke for this insight.

[180] UNIF. COMPUTER INFO. TRANSACTIONS ACT § 118(b) (amended 2002), 7 U.L.A. pt. 2, at 290 (2009 & Supp. 2010); *see also* Jonathan Band, *Closing the Interoperability Gap: NCCUSL's Adoption of a Reverse Engineering Exception in UCITA*, COMPUTER & INTERNET LAW., May 2002, at 1, 4.

[181] Maryland and Virginia have adopted the UCITA. *See* Maryland Uniform Computer Information Transactions Act, MD. CODE ANN., COM. LAW §§ 22-102 to -816 (West 2010); Uniform Computer Information Transactions Act, VA. CODE ANN. §§ 59.1-501.1 to -509.2 (West 2010).

[182] *Cisco Acts to Silence Researcher*, BBC NEWS, http://news.bbc.co.uk/2/hi/technology/4724791.stm (last updated July 28, 2005).

[183] Complaint for Misappropriation of Trade Secrets, Copyright, and Breach of Contract at 3–7, Cisco Sys., Inc. v. Lynn, No. C05-03043-JL (N.D. Cal. July 27, 2005).

[184] *See* ELDAD EILAM, REVERSING: SECRETS OF REVERSE ENGINEERING 7–8 (2005).

generally a prerequisite for using a lawful copy of that program. Researchers thus face a cruel choice: either use an unlawful copy in their research, or agree to terms preventing them from engaging in the activity that necessitates installing the program. To mitigate this problem, courts interpreting software license agreements, and state legislatures adopting and modifying trade secret statutes, should reinforce the position that reverse engineering does not constitute improper means.[185] Provisions banning reverse engineering could be voided on public policy grounds, for example. In addition, security research should be exempted from trade secret liability, unless the plaintiff can prove improper means.[186] An exemption would shift the burden of proof to the software's owner and would continue to protect against breaches of nondisclosure agreements. This proposal accords with the goal of trade secret doctrine, which is to enable the protection of proprietary information that confers a competitive advantage. Security researchers seek not to compete with the software they test, but to improve its resilience and robustness. Further, breach of an end-user license agreement should give rise, at most, to a claim for breach of contract. Unlike trade secret claims, contract-based ones rarely justify injunctive relief, and plaintiffs must prove actual damages to recover.[187]

Trade secret doctrine is created primarily by state law, although theft of trade secrets can create federal criminal liability.[188] Thus, ensuring uniform protection for security research requires either action by each state to protect reverse engineering, or a federal statute enshrining this shield nationally. While some states already safeguard reverse engineering via statute, researchers may nonetheless face restrictions based on trade secret law. Hoffman and Griffith, for example, were subject to a temporary restraining order based in part on Blackboard's trade secret claim,[189] even though

---

[185] Trade secrets statutes typically prohibit acquisition of protected information via improper means. Most states follow the formulation of the Uniform Trade Secrets Act, which defines *improper means* as including "theft, bribery, misrepresentation, breach or inducement of a breach of a duty to maintain secrecy, or espionage through electronic or other means." UNIF. TRADE SECRETS ACT § 1(1) (amended 1985), 14 U.L.A. 537 (2005). States such as California expressly protect reverse engineering as an exception to trade secret liability. *See* CAL. CIV. CODE § 3426.1 (West 2010) ("Reverse engineering or independent derivation alone shall not be considered improper means.").

[186] *Cf.* 17 U.S.C. § 1201(j) (2006) (DMCA security research exemption).

[187] *See* Doug Rendleman, *The Inadequate Remedy at Law Prerequisite for an Injunction*, 33 U. FLA. L. REV. 346, 348, 351 (1981).

[188] *See* 18 U.S.C. § 1832 (2006).

[189] *See* Order, *supra* note 154; Verified Complaint, *supra* note 157, at 6.

Georgia's trade secret statute exempts reverse engineering from liability.[190] Accordingly, even states that protect reverse engineering in theory may need stronger liability shields in practice. This suggests that, if states fail to accord adequate protection to software security research as a legitimate activity under their trade secret laws, Congress may need to pass safe harbor legislation that preempts conflicting state statutes. Though trade secret law is historically the province of state regulation, the Federal Economic Espionage Act of 1996 criminalizes the theft, copying, distribution, sale, or receipt of unlawfully acquired trade secrets.[191] If federal law can be employed to augment trade secret when necessary, it can (and should) be deployed to limit the doctrine when its effects are pernicious.

D. Trademark

Unsurprisingly, trademark has been the legal doctrine least frequently employed by vendors to control software security research. Researchers are normally careful to note that their work does not bear the imprimatur or approval of a software vendor, and any references to a product or service would likely fall under the nominative use exception to trademark liability.[192] Furthermore, hackers are not offering competing products or appealing to consumers in a way that could cause confusion about source.[193] Blackboard, for example, included a claim under federal trademark law (the Lanham Act) in its complaint against Hoffman and Griffith over their BTS work.[194] As the Chilling Effects project notes, though, their position was "far-fetched at best," since Hoffman was neither passing off his work as Blackboard's, nor implying endorsement by the company.[195] Researchers are at pains both to claim credit for their work, and to demonstrate when a vendor has failed to follow their independent advice.

Trademark law—at least, federal trademark law—likely does not require modification to protect security research. However, exemption from liability does involve thoughtful judicial application of doctrine. For example, the

---

190 *See supra* note 173.

191 *See* 18 U.S.C. § 1832.

192 *See, e.g.*, New Kids on the Block v. News Am. Publ'g, Inc., 971 F.2d 302, 307–08 (9th Cir. 1992) (describing nominative use).

193 *See* 15 U.S.C. § 1125(a) (2006) (prohibiting use of a mark that is likely to cause confusion about origin, sponsorship, or approval).

194 Verified Complaint, *supra* note 157, at 6–7.

195 Jennifer Jenkins, *Blackboard Erases Research Presentation with Cease-and-Desist, TRO*, CHILLING EFFECTS (Sept. 30, 2003), http://www.chillingeffects.org/weather.cgi?WeatherID=383.

nominative use defense (one species of trademark's fair use defense) should immunize researchers who use a software product's mark to denominate the code to which their findings apply.[196] If hackers are careful in how they describe their work, consumer confusion should be minimal if not nonexistent. Similarly, federal dilution law provides express protection for nominative and descriptive fair use (including use to criticize or comment on the mark's owner), for news commentary employing a mark, and for noncommercial use.[197] Courts in different circuits, though, apply the nominative use defense differently.[198] Some judges may be receptive to vendors' suggestions that the use of their marks implies sponsorship or approval of the security researcher's work, on a likelihood of confusion theory.[199] Researchers would improve their chances of successfully asserting a nominative fair use defense through steps that reduce the potential to confuse computer consumers, such as through disclaimer statements that expressly negate any connection between the hacker and the software vendor.[200]

While trademark law has seen limited use against security researchers, the doctrine's built-in safeguards suggest that legal reform may not be immediately necessary so long as they are conscientiously applied.

E. Digital Millennium Copyright Act (DMCA)

Though there are few court cases applying it to security research, the anticircumvention provisions of the Digital Millennium Copyright Act (DMCA) recur frequently as a threat employed against hackers. In 2002, Hewlett-Packard (HP) fulminated against SNOsoft's publishing code that permitted an attacker to gain root (administrator) privileges on HP's Tru64 Unix operating system via a buffer overflow exploit, characterizing it as a

---

196 *See New Kids on the Block*, 971 F.2d at 307–08 (discussing nominative use).

197 *See* 15 U.S.C. § 1125(c)(3).

198 *Compare* Century 21 Real Estate Corp. v. LendingTree, Inc., 425 F.3d 211, 232 (3d Cir. 2005) (positioning nominative fair use as affirmative defense), *with* Playboy Enters., Inc. v. Welles, 279 F.3d 796, 801 (9th Cir. 2002) (holding that a nominative use defense simply changes the likelihood of confusion methodology), *superseded by statute on other grounds*, Trademark Dilution Revision Act of 2006, Pub. L. No. 109-312, § 2, 120 Stat. 1730, 1730 (codified at 15 U.S.C. § 1125(c)), *as recognized in* Levi Strauss & Co. v. Abercrombie & Fitch Trading Co., 633 F.3d 1158 (9th Cir. 2011).

199 *See* 15 U.S.C. §§ 1114(1), 1125(a)(1)(A).

200 *Cf. Century 21 Real Estate*, 425 F.3d at 231 ("[A] disclaimer must be considered in determining whether the alleged infringer accurately portrayed the relationship that existed between plaintiff and defendant.").

DMCA violation.[201] Though HP invoked the specter of criminal sanctions for SNOsoft's post,[202] the researchers also faced civil liability.[203] HP had known about the vulnerability since 2001—a different researcher had posted a separate exploit that achieved root access—but had not issued a patch.[204] When HP Chief Executive Carly Fiorina and the company were inundated with complaints from researchers, reporters, and even HP employees, the firm retreated from its threats against SNOsoft.[205] Nonetheless, the incident prompted a number of attendees at the Black Hat conference that year to consider the possibility of reducing vulnerability sharing with vendors,[206] and HP stated that it would forgo legal threats if researchers would "reveal security threats using industry standard security practice."[207]

Similarly, in 2003, Princeton graduate student J. Alex Halderman (now a professor at the University of Michigan) analyzed MediaMax CD3, a copy protection scheme for music CDs from SunnComm.[208] SunnComm claimed that the program offered "a verifiable and commendable level of security," but Halderman found that computer users could evade its restrictions through the simple expedient of holding down the Shift key (thereby disabling Microsoft Windows' Autorun feature) when loading the CD.[209] Doing so kept the disc from loading a device driver that blocked users from copying music.[210] Users who allowed the CD to install the driver software could also disable it using instructions Halderman provided.[211]

---

201 Declan McCullagh, *Security Warning Draws DMCA Threat*, CNET NEWS (July 30, 2002, 4:48 PM), http://news.cnet.com/2100-1023-947325.html; *see also* Letter from Kent Ferson, Vice President, Hewlett-Packard Unix Sys. Unit, to Adriel T. Desautels, Founder, SNOsoft (July 29, 2002), *available at* http://www.politechbot.com/docs/hp.dmca.threat.073002.html.

202 McCullagh, *supra* note 201.

203 *See* 17 U.S.C. § 1203(a), (c) (2006); McCullagh, *supra* note 201.

204 McCullagh, *supra* note 201. The exploit code is available at http://packetstorm.linuxsecurity.com/0101-exploits/tru-64.su.c.

205 Kim Zetter, *HP, Bug-Hunters Declare Truce*, PCWORLD (Aug. 9, 2002, 6:00 PM), http://www.pcworld.com/article/103853/hp_bughunters_declare_truce.html.

206 *Id.*

207 Declan McCullagh, *HP Backs Down on DMCA Warning*, ZDNET UK (Aug. 2, 2002, 7:45 AM), http://news.zdnet.co.uk/itmanagement/0,1000000308,2120211,00.htm (quoting HP General Manager Martin Fink) (internal quotation mark omitted).

208 J. Alex Halderman, *Analysis of the MediaMax CD3 Copy-Prevention System*, COMPUTER SCI. & ENG'G AT THE UNIV. OF MICH. (Nov. 13, 2004), http://www.cse.umich.edu/~jhalderm/pub/cd3.

209 *Id. See generally* Brij Khurana, *Halderman GS Sees Copy-Protection Flaw in New CDs*, DAILY PRINCETONIAN, Oct. 9, 2003, http://www.dailyprincetonian.com/2003/10/09/8785 (internal quotation marks omitted).

210 Halderman, *supra* note 208.

211 Khurana, *supra* note 209.

Halderman's work had a significant effect—SunnComm's stock dropped in value by $10 million in the days after its release.[212] SunnComm responded. The company released a statement indicating that it would sue Halderman for violating the DMCA and would refer the matter to federal law enforcement for possible criminal proceedings.[213] The company specifically cited his paper (published on Halderman's website) as "'disseminated in a manner which facilitates infringement' in violation of the DMCA or other applicable law,"[214] calling it potentially "a felony."[215] The company later acknowledged the potential chilling effect of such lawsuits on academic security research.[216]

Both HP and SunnComm rescinded their threats after a wave of unfavorable publicity. But the threat of suit under the DMCA has impeded research into software vulnerabilities, even by academics at major universities.[217] Halderman and Ed Felten, his advisor, delayed publishing data about security flaws in the Sony BMG copy protection system for CDs for a month while consulting counsel about managing DMCA risk.[218] While they did so, consumers of Sony music CDs remained vulnerable to hackers who could use a flaw in the anti-copying software to surreptitiously install software on their computers.[219] (Felten was familiar with DMCA threats, having faced one from the Secure Digital Music Initiative when he cracked the group's music watermarking scheme and sought to present his research at an academic

---

212 John Borland, *Student Faces Suit over Key to CD Locks*, CNET NEWS (Oct. 9, 2003, 2:01 PM), http://news.cnet.com/2100-1025-5089168.html.

213 Tony Smith, *SunnComm to Sue 'Shift Key' Student for $10m*, REGISTER (Oct. 9, 2003, 8:48 PM), http://www.theregister.co.uk/2003/10/09/sunncomm_to_sue_shift_key.

214 *Id.* (quoting a SunnComm press release) (internal quotation mark omitted).

215 Declan McCullagh, *SunnComm Won't Sue Grad Student*, ZDNET (Oct. 10, 2003, 9:16 PM), http://news.zdnet.com/2100-3513_22-132123.html.

216 Press Release, SunnComm Techs. Inc., SunnComm Technologies Reverses Decision to Bring Legal Action Against Princeton Researcher (Oct. 10, 2003), http://web.archive.org/web/20031016165917/http://www.sunncomm.com/press/pressrelease.asp?prid=200310101150.

217 *See generally* FRED VON LOHMANN, ELEC. FRONTIER FOUND., UNINTENDED CONSEQUENCES: TWELVE YEARS UNDER THE DMCA (2010), *available at* http://www.eff.org/files/eff-unintended-consequences-12-years.pdf; J. Alex Halderman, Princeton Univ., Legal Challenges in Security Research (Oct. 12, 2006) (PowerPoint presentation), *available at* http://www.cse.umich.edu/~jhalderm/pub/talks/lawsec-ufl06.ppt.

218 Comment of Edward W. Felten & J. Alex Halderman to the U.S. Copyright Office, Regarding RM 2005-11—Exemption to Prohibition on Circumvention of Copyright Protection Systems for Access Control Technologies (Dec. 1, 2005), *available at* http://www.freedom-to-tinker.com/doc/2005/dmcacomment.pdf.

219 The Sony BMG software employed a rootkit—a storage space invisible to the operating system—to hide its tools. Mark Russinovich, *Sony, Rootkits and Digital Rights Management Gone Too Far*, MARK'S BLOG (Oct. 31, 2005, 11:04 AM), http://blogs.technet.com/markrussinovich/archive/2005/10/31/sony-rootkits-and-digital-rights-management-gone-too-far.aspx. However, attackers could also conceal their software in the rootkit. *See* John Borland, *Sony CD Protection Sparks Security Concerns*, CNET NEWS (Nov. 1, 2005, 4:41 PM), http://news.cnet.com/Sony-CD-protection-sparks-security-concerns/2100-7355_3-5926657.html.

conference.)[220] Andrew "Bunnie" Huang had two companies—including a self-publishing firm—back out of publishing his book on hacking Microsoft's Xbox (including analysis of the system's security) due to fears of DMCA liability.[221] University of Michigan graduate student Niels Provos moved his research publications out of the United States and tried to block American citizens from accessing them due to fears of running afoul of the DMCA and a similar Michigan state statute.[222] Even White House Office of Cyber Security chief Richard Clarke cited the "potential chilling effect on vulnerability research" in a speech at the Massachusetts Institute of Technology.[223] The DMCA has proved a potent weapon, enabling software companies to dissuade or limit security researchers, and its power is perhaps best demonstrated by its ability to compel adherence even with infrequent formal legal proceedings.

The DMCA contains statutory exceptions that could shield security researchers from liability, including protections for reverse engineering,[224] encryption research,[225] and security testing.[226] However, the safe harbors are so narrow that they are effectively useless, as the extant caselaw demonstrates. Of 141 decided cases involving § 1201 of the DMCA, only one involved a claim of protection under the security testing safe harbor, and in it the safe harbor was held inapplicable.[227] The same case was the only one to involve an unsuccessful attempt to rely on the encryption research exemption,[228] and four cases had unsuccessful claims for the reverse engineering safe harbor.[229] While these results cover only reported, decided cases (and hence may not be a representative sample), the lack of success in using any of the safe harbors—

---

220 *See* Scott A. Craver et al., *Reading Between the Lines: Lessons from the SDMI Challenge*, 10 PROC. USENIX SECURITY SYMP. (2001), *available at* http://www.usenix.org/events/sec01/craver.pdf.

221 David Becker, *Testing Microsoft and the DMCA*, CNET NEWS (Apr. 15, 2003, 4:00 AM), http://news.cnet.com/2008-1082-996787.html (quoting Huang's description of "a flaw in the system initializer that lets you put code anywhere in the system that you want it").

222 Kevin Poulsen, *'Super-DMCA' Fears Suppress Security Research*, SECURITYFOCUS (Apr. 14, 2003), http://www.securityfocus.com/news/3912.

223 Hiawatha Bray, *Cyber Chief Speaks on Data Network Security*, BOS. GLOBE, Oct. 17, 2002, at C2.

224 17 U.S.C. § 1201(f) (2006).

225 *Id.* § 1201(g).

226 *Id.* § 1201(j).

227 Universal City Studios, Inc. v. Reimerdes, 111 F. Supp. 2d 294 (S.D.N.Y. 2000), *aff'd sub nom.* Universal City Studios, Inc. v. Corley, 273 F.3d 429 (2d Cir. 2001).

228 *Id.*

229 Davidson & Assocs. v. Jung, 422 F.3d 630 (8th Cir. 2005); Lexmark Int'l, Inc. v. Static Control Components, Inc., 387 F.3d 522 (6th Cir. 2004); Sony Computer Entm't Am., Inc. v. Divineo, Inc., 457 F. Supp. 2d 957 (N.D. Cal. 2006); *Reimerdes*, 111 F. Supp. 2d 294.

and the infrequency with which they are raised—suggests that the built-in statutory mechanisms are insufficiently protective of security researchers.

This conclusion is bolstered by qualitative analysis of the exemptions. To qualify for the security testing safe harbor, a researcher must: obtain authorization from the owner of the computer, system, or network involved in testing (which might be particularly challenging for cloud-computing research); not violate the Computer Fraud and Abuse Act (CFAA); use findings solely to improve the security of the computer, system, or network's owner, or share them directly with the network, system, or computer's developer; and use or maintain the information derived from testing so as not to facilitate DMCA infringement or violate other applicable laws, such as those related to privacy and security.[230] Similarly, to assess vulnerabilities in software encryption, such as that employed in the Transport Layer Security protocol used to protect e-commerce,[231] a researcher must lawfully obtain a copy of the software or hardware, seek authorization from the owner of the rights in that technology, and not violate other laws (including the CFAA).[232] Moreover, the statute conditions the exemption on the researcher's qualifications, the way in which she disseminates her findings, and whether the researcher provides the copyright owner with documentation of findings in a timely fashion.[233] The statutory safe harbors are not only narrow, but also uncertain—it is not always clear what conduct violates laws such as the CFAA, nor what constitutes timely provision of information to copyright owners.[234]

The DMCA permits users who are adversely affected by its restrictions in their ability to make noninfringing uses of copyrighted works to petition the Librarian of Congress to exempt certain classes of works from the statute's ambit.[235] However, these exemptions expire after three years, and a user who seeks to continue the exemption must petition for renewal from scratch.[236] The first three rounds of exemption rule making did not result in additional

---

230 17 U.S.C. § 1201(j)(1)–(3).

231 *See* Marsh Ray, *Authentication Gap in TLS Renegotiation*, EXTENDED SUBSET (Nov. 5, 2009, 3:20 AM), http://extendedsubset.com/?p=8.

232 17 U.S.C. § 1201(g)(2).

233 *Id.* § 1201(g)(3).

234 On the Computer Fraud and Abuse Act (CFAA), see generally *United States v. Drew*, 259 F.R.D. 449 (C.D. Cal. 2009), in which a court granted a motion for acquittal for a defendant convicted of violating CFAA via conduct that contravened MySpace's terms of service.

235 17 U.S.C. § 1201(a)(1)(C)–(D).

236 Exemption to Prohibition on Circumvention of Copyright Protection Systems for Access Control Technologies, 73 Fed. Reg. 58,073 (Oct. 6, 2008) (codified as amended at 37 C.F.R. pt. 201).

protections that could benefit security researchers.[237] The last round of rule making included one exemption for security testing of video games, provided the information derived is used primarily to promote the security of the owner or operator of a computer, computer system, or network, and the information is not used or maintained such that it facilitates unlawful activity.[238] This exemption is quite similar to the existing statutory exemption for security testing, but is intended to cover situations where the researcher is not trying to gain access to a computer, system, or network.[239] In short, while it is possible for adversely affected users to petition for exemptions from liability, such exemptions are narrow, short-lived, and vigorously opposed by vendors.[240]

Reform of the DMCA to provide greater protection for software security research is straightforward. As currently written, the statute canonizes one type, or industry structure, for such research.[241] It requires that, to fall within the safe harbor for security testing, a researcher must perform her activities with the authorization of the owner or operator of the system or network.[242] Thus, the DMCA protects research where the investigator operates under contract with the software vendor or cloud-computing system operator, but leaves independent researchers vulnerable. This type of research is carried out by corporate security firms such as Verisign iDefense Labs, Defensive Thinking, or Symantec. Individual researchers or smaller companies may have trouble obtaining authorization due to negotiation costs or because software firms simply may not trust them.

To change the DMCA to more broadly protect the activity of security research, rather than simply one organizational form of it, Congress should either amend the relevant statutory subsection, or simply treat the DMCA under a more generalized shield law. To carry out piecemeal reform, the DMCA should focus on the activity of the security researcher, not on purpose or on authorization. (Ironically, the current statutory exemption implicitly recognizes the key role of the distribution of vulnerability information—it conditions the safe harbor in part on whether the researcher shares the data

---

237 *Anticircumvention Rulemaking*, U.S. COPYRIGHT OFFICE, http://www.copyright.gov/1201 (last updated Feb. 7, 2011).

238 37 C.F.R. § 201.40(b)(4) (2010).

239 75 Fed. Reg. 43,825, 43,833 (July 27, 2010) (codified as amended at 37 C.F.R. § 201.40).

240 *See id.* at 43,832–33 (documenting arguments of opponents to video game security exemption).

241 *See generally* Cross, *supra* note 25, at 39 (noting that vital security tools, such as Nmap, NetCat, and OllyDbg, were developed by independent researchers).

242 17 U.S.C. § 1201(j)(1) (2006).

directly with the computer system owner or software developer.)[243] In particular, we suggest removing the requirement of obtaining authorization from the owner of the computer, system, or network. Overall, though, we believe that more comprehensive reform, which treats the DMCA as one aspect of IP problems facing security researchers, is preferable.

The DMCA illustrates the potency of intellectual property threats to security research by showing how dissemination of information can be chilled even without filing suit. This Part has demonstrated the tools available to software vendors to muzzle researchers. The next Part describes how to mitigate this problem.

III. CREATING THE AEGIS

To protect researchers' valuable contributions to software security, and to ensure that vulnerability information remains in legitimate channels rather than being sold on the black market, we propose three changes: one legal, one in social norms, and one market-based.

A. Legal Reform

Our proposed legal reform seeks to shape researchers' behavior by conditioning a grant of immunity from IP suits on adherence to rules of conduct. Providing a safe harbor from liability strongly encourages those at risk to act in ways that remain within the exemption. For example, the Digital Millennium Copyright Act provides a safe harbor for service providers who, upon notice from a copyright holder, disable access to or remove allegedly infringing content.[244] Most large providers, as a matter of course, remove content upon notification without inquiring into the merits of the alleged infringement claim or into their potential risk exposure.[245] While this approach

---

[243] *See id.* § 1201(j)(3) (stating that one factor to determine whether one qualifies for the exemption is "whether the information derived from the security testing was . . . shared directly with the developer of such computer, computer system, or computer network").

[244] *Id.* § 512(c).

[245] *See, e.g.*, Michael Piatek et al., *Challenges and Directions for Monitoring P2P File Sharing Networks—or—Why My Printer Received a DMCA Takedown Notice*, 3 USENIX WORKSHOP ON HOT TOPICS SECURITY (2008), *available at* http://www.usenix.org/event/hotsec08/tech/full_papers/piatek/piatek.pdf; Jennifer M. Urban & Laura Quilter, *Efficient Process or "Chilling Effects"? Takedown Notices Under Section 512 of the Digital Millennium Copyright Act*, 22 SANTA CLARA COMPUTER & HIGH TECH. L.J. 621 (2006).

has generated controversy, it is more certain and less expensive to stay within the safe harbor.[246]

Similarly, if software security researchers can avail themselves of immunity from IP infringement claims by acting in certain ways, it is likely they will conform their behavior to those requirements. Many researchers have limited resources and legal acumen, making them risk-averse regarding litigation and thus more likely to track the exemption's mandates. Legal threats unquestionably influence researchers' actions, as they learn from prior controversies. For example, a trio of security experts demonstrated how to hack smartcard-based electronic parking meters at the Black Hat security conference in August 2009.[247] The researchers deliberately chose to contact neither the vendor nor the owner of the meters they hacked, and asked reporters not to do so, citing the injunction entered against MIT researchers who showed how to obtain free rides on Boston's subway system in 2008.[248] Hackers learn the relevant law quickly.

Crafting the requirements for immunity is critical in two respects. First, conditions for the exemption will strongly influence how hackers behave—what they do while testing code, and what they do with the resulting information.[249] Second, the safe harbor would deprive software vendors of potent legal tools and remedies. If it immunizes undesirable behavior, it will inflict harm on software firms, and on society generally.

Proper behavior for researchers is admittedly a contested issue within the software security community.[250] Proposals vary from advocating full disclosure[251] (publishing vulnerability details immediately upon discovery) to

---

246 *See, e.g.*, Peter Lattman, *Law Professor Wendy Seltzer Takes on the NFL*, WALL ST. J. L. BLOG (Mar. 21, 2007, 12:27 PM), http://blogs.wsj.com/law/2007/03/21/law-professor-wendy-seltzer-takes-on-the-nfl/tab/article.

247 Joe Grand et al., "Smart" Parking Meter Implementations, Globalism, and You (2009) (PowerPoint presentation), http://www.blackhat.com/presentations/bh-usa-09/GRAND/BHUSA09-Grand-ParkingMeter-SLIDES.pdf.

248 Kim Zetter, *Smart Parking Meters Hacked—Free Parking for All!*, WIRED THREAT LEVEL (July 30, 2009, 4:51 PM), http://www.wired.com/threatlevel/2009/07/parking-meters.

249 The proposed legal reform can therefore be seen as attempting to specify best practices for security research. We are grateful to Michael J. Madison for this insight; he develops a similar idea for the fair use doctrine under copyright law in Michael J. Madison, *A Pattern-Oriented Approach to Fair Use*, 45 WM. & MARY L. REV. 1525 (2004).

250 *See generally* VIEGA, *supra* note 38, at 153–62.

251 Bruce Schneier, *Debating Full Disclosure*, SCHNEIER ON SECURITY (Jan. 23, 2007, 6:45 AM), http://www.schneier.com/blog/archives/2007/01/debating_full_d.html ("Public scrutiny is the only reliable way to improve security . . . .").

revealing bug data only to vendors.[252] Heated debate is common[253] and occasionally bleeds into active protests such as website hacking.[254] The normative position we adopt, responsible disclosure, represents a middle ground that has won considerable support.[255] Responsible disclosure requires researchers to notify vendors first on discovering vulnerabilities, but preserves the possibility of public dissemination to prod software firms to remediate flaws.[256] We believe that the potential for full public disclosure under our proposal motivates vendors to issue patches and to press customers to install those fixes, while the prohibition on selling vulnerability data to third parties reduces the number of potential attackers until the vulnerability can be remedied.[257]

The Hacker's Aegis would set a default presumption that security researchers are acting lawfully, and would require plaintiffs (such as aggrieved software companies) to demonstrate that accused activity falls outside its protections. In return for immunity from civil intellectual property claims, software security researchers would be required to adhere to five rules: tell the vendor first, don't sell the bug, test on your own system, don't weaponize, and create a trail.

---

[252] The Anti-Sec Movement, for example, opposes full disclosure "for the purpose of making it harder for the security industry to exploit its consequences." *See ImageShack—Pwned for Anti-Sec*, SECLISTS.ORG (July 11, 2009, 5:15 AM), http://seclists.org/fulldisclosure/2009/Jul/95.

[253] *See, e.g.*, Marcus J. Ranum, *The Vulnerability Disclosure Game: Are We More Secure?*, CSO ONLINE (Mar. 1, 2008), http://www.csoonline.com/article/440110/The_Vulnerability_Disclosure_Game_Are_We_More_Secure_?CID=28073; Bruce Schneier, *Schneier: Full Disclosure of Security Vulnerabilities a 'Damned Good Idea,'* CSO ONLINE (Jan. 9, 2007), http://www.csoonline.com/article/216205/schneier-full-disclosure-of-security-vulnerabilities-a-damned-good-idea-.

[254] *See, e.g.*, John Leyden, *ImageShack Hacked in Oddball Security Protest*, REGISTER (July 13, 2009), http://www.theregister.co.uk/2009/07/13/imageshack_hack (describing defacement of ImageShack site by Anti-Sec, a group protesting full disclosure).

[255] *See, e.g.*, STEPHEN A. SHEPHERD, SANS INST., VULNERABILITY DISCLOSURE: HOW DO WE DEFINE RESPONSIBLE DISCLOSURE? (2003), *available at* http://www.sans.org/reading_room/whitepapers/threats/how_do_we_define_responsible_disclosure_932?show=932.php&cat=threats; Chris Evans et al., *Rebooting Responsible Disclosure: A Focus on Protecting End Users*, GOOGLE ONLINE SECURITY BLOG (July 20, 2010, 2:07 PM), http://googleonlinesecurity.blogspot.com/2010/07/rebooting-responsible-disclosure-focus.html; *Vulnerability Disclosure Publications and Discussion Tracking*, OULU UNIV. SECURE PROGRAMMING GRP., https://www.ee.oulu.fi/research/ouspg/Disclosure_tracking (last visited May 13, 2011) (tracking debate). However, Google has recently shifted its stance, albeit slightly, focusing on reasonable disclosure deadlines rather than responsible disclosure. *See Google Security and Product Safety*, GOOGLE, http://www.google.com/corporate/security.html (last visited May 13, 2011).

[256] *See, e.g.*, Steve Christey & Chris Wysopal, Responsible Vulnerability Disclosure Process (Feb. 2002) (unpublished internet draft), http://www.wiretrip.net/rfp/txt/ietf-draft.txt.

[257] *But see* Brad Spengler, *Hyenas of the Security Industry*, SECLISTS.ORG (June 18, 2010, 1:19 AM), http://seclists.org/dailydave/2010/q2/58 (criticizing responsible disclosure and vendor lag in patching bugs).

1. Tell the Vendor First

The first conduct-based rule would require a researcher who discovers a security vulnerability to report it to the vendor of the affected software before publishing any information about the flaw. Failure to report before disclosing information about the bug would bar the researcher from availing herself of the safe harbor, but should not function as evidence of infringement or any other legal liability. Hackers should tell the software producer—the party best positioned to remedy the flaw—first.

The reporting requirement includes two additional components. The first would mandate that researchers use the method of contact described on the home page of a vendor's site; if the vendor fails to include a contact mechanism on their site, the researcher may simply notify customer support or the firm's general counsel.[258] Bugs submitted to tech support or to a company's lawyer are less likely to receive attention, which is why this approach seeks to press vendors to establish a means of gathering vulnerability data. Companies would likely opt to create such mechanisms because failure to do so would let researchers potentially avail themselves of the safe harbor merely by contacting technical support. Support representatives may not be trained to deal properly with security reports, and hence firms would lose legal recourse without much corresponding benefit. Researchers would use the designated contact path because immunity depends upon it. Moreover, hackers want bugs to be taken seriously, and sending findings into proper channels increases the likelihood that that will occur. A contact system modeled on the designated agent to receive notifications of claimed copyright infringement under the Digital Millennium Copyright Act would be optimal.[259] The DMCA confers immunity upon online service providers who make available on their websites contact information for this agent; copyright owners are increasingly accustomed to looking on a service provider's home page for contact details.[260] The contact mechanism would reduce search and communication costs for both parties.

The second component would mandate a postreporting delay before the researcher could share vulnerability data publicly. This would provide time for

---

[258] *See, e.g.*, *Bug Reporting*, APPLE DEVELOPER, http://developer.apple.com/bugreporter (last visited May 13, 2011); *Report a Microsoft Product Bug*, MICROSOFT SUPPORT, http://support.microsoft.com/gp/contactbug (last updated July 30, 2009).

[259] *See Service Provider Agents*, U.S. COPYRIGHT OFFICE, http://www.copyright.gov/onlinesp/list/a_agents.html (last visited May 13, 2011).

[260] *See* 17 U.S.C. § 512(c)(2) (2006).

the vendor to assess the new information, contact the researcher, and begin work upon a patch if necessary.[261] Public disclosure before this period would negate the safe harbor. After the initial waiting period passes, the researcher would be free to share the vulnerability data.[262] Researchers interested in blackmail are unlikely to follow the safe harbor's rules in any case, and public disclosure is an important incentive to compel vendors to take bugs seriously. Trusted intermediaries, such as the Computer Emergency Response Team Communication Center (CERT/CC), disclose vulnerabilities publicly forty-five days after they are initially reported, regardless of the status of patches from vendors.[263] The term of the delay period must balance providing vendors with sufficient time to patch, against the risk to users from unpatched vulnerabilities and the need to press software manufacturers to act promptly. This is ultimately an empirical question, but we believe the CERT/CC forty-five-day model is a useful starting point—particularly since vendors and the security community are already accustomed to it.

2. Do Not Sell the Bug

The second behavior rule would ban sales of vulnerability data to third parties. Researchers would forfeit the safe harbor if there were sufficient credible evidence that they offered data about the vulnerability to any third party for compensation. (This would permit transactions with the vendor.) By "sufficient credible evidence," we mean concrete facts, and not conclusory allegations or statements on information and belief by vendors. To defeat immunity based on this factor, a software provider would have to adduce and support facts sufficient to survive a motion to dismiss.[264] While this might seem to impair programs such as Tipping Point's Zero Day Initiative, where independent firms pay researchers for reporting security bugs to them, this concern is readily mitigated via private law.[265] If participating in third-party "bounty hunter" programs makes a researcher ineligible for the public law safe harbor, researchers will either demand that their agreements with these

---

[261] *See generally Coordinated Vulnerability Disclosure*, MICROSOFT SECURITY RESPONSE CENTER, http://www.microsoft.com/security/msrc/report/disclosure.aspx (last visited May 13, 2011).

[262] *See generally* VIEGA, *supra* note 38, at 153–62.

[263] *CERT/CC Vulnerability Disclosure Policy*, COMPUTER EMERGENCY RESPONSE TEAM, http://www.cert.org/kb/vul_disclosure.html (last updated May 15, 2008).

[264] The weight of evidence required to overcome a motion to dismiss is unclear after the Supreme Court's decisions in *Ashcroft v. Iqbal*, 129 S. Ct. 1937 (2009), and *Bell Atlantic Corp. v. Twombly*, 550 U.S. 544 (2007). We do not believe that these shifts in the standard will have a material effect on the safe harbor.

[265] *See* ZERO DAY INITIATIVE, http://www.zerodayinitiative.com (last visited May 13, 2011).

programs indemnify them against legal risks from IP suits, thus effectively reproducing the immunity, or the price that they require for reporting bugs will increase to compensate them for the risk.[266] The goal of this factor is to discourage researchers from engaging in strategic behavior by marketing their wares to the underground before, or concurrently with, offering them to vendors. While proof of efforts to sell vulnerability data may be difficult to obtain, we believe that the existence of this factor will help to discourage gray hat hackers—those willing to act as black hats or white hats depending on circumstances—from selling their findings to anyone other than the vendor.

3. Test on Your Own System

Third, researchers must test for vulnerabilities on their own systems unless they cannot reasonably do so, as with cloud computing. A researcher would lose the safe harbor on a showing of sufficient credible evidence that she tested, or employed the flaw to compromise security on, a system not under her lawful control, unless there was no reasonable alternative. This factor is intended to push researchers to investigate vulnerabilities on test systems rather than on production code that is in use by others.[267] Analysis of whether there is a reasonable alternative should be searching; hackers should not interfere with others' systems lightly. For example, if a software company employs only open source code,[268] or makes trial versions of its products available,[269] a researcher would have ready access to the relevant code. Hence, the researcher would have to test on her own system to stay within the safe harbor.

However, the rise of web services and cloud computing complicates this analysis. In cases where the software is available only from third-party systems, such as with Amazon's Elastic Compute Cloud (EC2),[270] researchers should be permitted to test vulnerabilities hosted on those systems as long as the researcher does not use the system to do more than verify the existence and extent of the vulnerability, and does not cause more than temporary, minor

---

[266] *Cf.* Gary T. Schwartz, *The Ethics and the Economics of Tort Liability Insurance*, 75 CORNELL L. REV. 313, 351 n.165 (1990) (discussing *ex ante* risk compensation in tort).

[267] *See, e.g.*, *Amazon Elastic Compute Cloud (Amazon EC2)*, AMAZON WEB SERVICES, http://aws.amazon.com/ec2 (last visited May 13, 2011); *Virtualization Security Tops RSA 2010 Innovation Sandbox*, WIKIBON BLOG (Mar. 2, 2010), http://wikibon.org/blog/virtualization-security-tops-rsa-2010-innovation-sandbox.

[268] *See, e.g.*, *Chromium*, GOOGLE CODE, http://code.google.com/chromium (last visited May 13, 2011).

[269] *See, e.g.*, *Free Trial Software*, MCAFEE, http://home.mcafee.com/Store/FreeTrial.aspx (last visited May 13, 2011). This might also helpfully push vendors to make versions available for testing.

[270] *Amazon Elastic Compute Cloud (Amazon EC2)*, *supra* note 267.

disruption to the operation of the service. While this qualification makes this third factor closer to a standard than a rule for cloud computing, it is necessary to enable research on "software as a service" applications and to protect researchers against claims that their testing harmed or impeded the service.[271]

4. Do Not Weaponize

Fourth, the researcher must not publish, without the vendor's authorization, exploit- or proof-of-concept code that enables attacks against the vulnerability. Researchers who "weaponize" vulnerabilities increase the number of potential attackers. Descriptions of security flaws may allow sophisticated black hats to create programs that leverage bugs, but tools that automatically attack weaknesses allow *any* user who downloads them to wreak havoc.[272] While exploit code may alert system administrators to methods of protecting against vulnerabilities, the risk of attacks from "script kiddies" outweighs the gain in safety.[273] If a vendor can show sufficient credible evidence that the researcher has published weaponized code, the researcher would forfeit immunity.

5. Create a Trail

Finally, the researcher must create an audit trail for the vulnerability by reporting it to the clearinghouse described below in section D. To qualify for the safe harbor, the discoverer of a bug must upload a detailed description of the flaw, information on how to reproduce it, any known exploits or proof-of-concept code, her identifying information, and a copy of any correspondence (such as e-mail) with the vendor. This provides proof that a researcher found and elucidated a bug, and that she provided the vendor with sufficient information to investigate it. Moreover, mandating that researchers supply contact information enables vendors to communicate with them, and also deters strategic behavior, such as claiming credit for others' discoveries.

The safe harbor for researchers who follow these five rules should be structured as an exemption from liability and not merely as a defense. The

---

[271] *See* Christina Torode, *Cost-Effective Web Application Security Testing Options Take SaaS Form*, SEARCHCIO-MIDMARKET.COM (Dec. 17, 2009), http://searchcio-midmarket.techtarget.com/tip/0,289483,sid183_gci1377140,00.html. On standards versus rules in security, see Derek E. Bambauer, *Rules, Standards, and Geeks*, 5 BROOK. J. CORP. FIN. & COM. L. 49 (2010).

[272] *See* Courtlend Little, *Weaponization Trumps Skill*, SC MAG. (Aug. 14, 2008), http://www.scmagazineus.com/weaponization-trumps-skill/article/115432.

[273] *See generally* Andy Greenberg, *The No-Tech Hacker*, FORBES.COM (Feb. 29, 2008, 6:00 AM), http://www.forbes.com/2008/02/28/long-hacker-csc-tech-security-cx_ag_0229hacker.html.

difference between an exemption and a defense can be seen by comparing copyright's fair use defense to the exemption from liability for third-party speech under § 230 of the Communications Decency Act.[274] Fair use's burden falls upon the defendant; the plaintiff need not prove that the alleged infringement was unfair.[275] By contrast, a plaintiff alleging, for example, that a website is responsible for comments posted by third-party users must show that the site falls outside the exemption from liability created by § 230.[276] An exemption is preferable for three reasons. First, the safe harbor is most relevant, and important, in the initial stages of a dispute between a researcher and a vendor. Few IP lawsuits against hackers go to trial. Mike Lynn settled with Cisco,[277] as Griffith and Hoffman did with Blackboard.[278] The goal of a suit is not to win in court, but to prevent publication or use of vulnerability data by the researcher, to gain time for the vendor to respond to the bug (both in creating patches and in managing public perception of its product), and to force the researcher to agree to terms favorable to the company in settling the dispute. The first stage of the fight is typically a request for a temporary restraining order or preliminary injunction that constrains the hacker's options and conduct.[279] Often, requests for such an order are handled ex parte, as with Mike Lynn and Billy Hoffman. Thus, whether the researcher has an applicable defense is irrelevant, unless it is sufficiently strong to affect the judge's perception of whether the plaintiff will succeed on the merits of its claim.[280] (This is unlikely, given that the vendor controls what evidence and arguments are adduced in the initial hearing.) Once the vendor has the order, the researcher must move to change the status quo.

Second, structuring the safe harbor as a defense would likely not reduce the costs researchers face sufficiently. Hackers would have to muster evidence

---

274 47 U.S.C. § 230 (2006).

275 *See, e.g.*, Campbell v. Acuff-Rose Music, Inc., 510 U.S. 569, 574 (1994) ("It [was] uncontested here that [defendant's] song would be an infringement of [plaintiff's] rights in [the copyrighted work] under the Copyright Act of 1976, but for a finding of fair use . . . ." (citation omitted)).

276 *See, e.g.*, Doe v. MySpace, Inc., 528 F.3d 413, 422 (5th Cir. 2008) (finding no liability). *But see* Fair Hous. Council v. Roommates.com, LLC, 521 F.3d 1157, 1162–63, 1166 (9th Cir. 2008) (finding liability where site's design made it partially responsible for content).

277 Brian Krebs, *Text of the Cisco-ISS-Lynn-Black Hat Agreement*, WASH. POST SECURITY FIX (July 29, 2005, 12:35 PM), http://blog.washingtonpost.com/securityfix/2005/07/text_of_the_ciscoisslynnblack.html.

278 *Blackboard Reaches Settlement with Hacker Duo*, RFIDNEWS (July 16, 2003), http://www.rfidnews.org/2003/07/16/blackboard-reaches-settlement-with-hacker-duo.

279 *See, e.g.*, Verified Complaint, *supra* note 157.

280 *See* Winter v. Natural Res. Def. Council, Inc., 129 S. Ct. 365, 374 (2008) (citing standard for preliminary injunction).

and legal arguments to support their eligibility for the exemption.[281] Even hiring counsel can be an expensive proposition for individual researchers.

Finally, altering the allocation of the burden of proof is important to this Article's larger normative goals. Given its social utility, security research should be presumptively legitimate, not unlawful. It should be incumbent upon an aggrieved vendor to overcome this presumption of legality, not for researchers to validate their activities.

Thus, the legal reform component of the Hacker's Aegis would establish exemption from IP-based liability for researchers who follow five rules: alert the vendor first; do not offer to sell data to any third party; refrain from testing systems not under their control unless there is no reasonable alternative; do not weaponize code; and create an audit trail. To break through this exemption, vendors would have to show facts that demonstrate that the researcher has violated one of these five requirements.

B. Form for Substance

Legal reform to shield security researchers from the threat of IP litigation by vendors could follow one of two paths. The first initiates new legislation to exempt research from liability. The second adapts existing doctrinal defenses in IP to cover researchers' activities. Each approach confers benefits, and faces challenges. Overall, we believe research-specific legislation is the preferable path.

A statute conferring immunity upon a designated class of actors—security researchers—has several advantages.[282] First, legislation could tailor the exemption to reward helpful behavior while leaving malefactors at risk of liability. Second, actor-specific rules could eliminate strategic behavior by vendors and others alleging infringement. If protections for researchers varied by IP doctrine, aggrieved software companies would seek to frame their claims under the theory with the narrowest protection.[283] This is particularly applicable for software, which can be protected under multiple, overlapping IP regimes. Third, it likely operates more rapidly than an accretion of doctrine-

---

[281] *Cf.* Campbell v. Acuff-Rose Music, Inc., 510 U.S. 569, 590 (1994) ("Since fair use is an affirmative defense [to a claim of copyright infringement], its proponent would have difficulty carrying the burden of demonstrating fair use without favorable evidence . . . ." (footnote omitted)).

[282] *See generally* Roscoe Pound, *Common Law and Legislation*, 21 HARV. L. REV. 383 (1908).

[283] *See, e.g.*, Complaint, *supra* note 183 (adducing multiple IP theories for liability); Verified Complaint, *supra* note 157.

specific exemptions developed from individual cases. A specific law that operated uniformly across IP doctrines and across jurisdictions would provide a more complete and more rapid shield than case-by-case development.

However, employing a statute specific to vulnerability research also has weaknesses. Most importantly, public choice problems make it likely that such a law would be underprotective.[284] Owners of intellectual property in software are concentrated and relatively powerful.[285] They have strong financial incentives to maximize IP protection for their code and would likely oppose, or seek to weaken, a research exemption. Other powerful interests—for example, those whose content is protected by code, such as the movie industry—would likely side with vendors.[286] By contrast, independent researchers tend to be individuals or small firms with less political sophistication and fewer resources. The situation is analogous to the ecosystem of interests involved in crafting copyright legislation described by Jessica Litman: content owners are politically sophisticated, resourceful, and have significant stakes in the outcome, while users and public interests are weaker, dispersed, and lack an effective representative.[287] A law protecting vulnerability research might appear (if at all) like the exemptions under the DMCA for security research, encryption research, and reverse engineering, which are so narrow that they have only been advanced in five cases since 1998, and never successfully.[288] Moreover, weak protection might be worse for researchers than none at all, as it would be difficult to argue that their actions should be protected if they fell outside the scope of legislatively determined permissible behavior.

The other option—employing doctrine-specific exceptions to protect security research—also confers benefits. It has the standard virtues of common law adjudication: judges can adapt protections to fit different circumstances, and variation among courts permits helpful experimentation in

---

284 *See generally* LANDES & POSNER, *supra* note 17, at 403–04, 416.

285 *See generally* William W. Fisher III, *The Growth of Intellectual Property: A History of the Ownership of Ideas in the United States*, *in* 1 INTELLECTUAL PROPERTY RIGHTS 72, 82–84 (David Vaver ed., 2006).

286 *See, e.g.*, *Content Protection*, MOTION PICTURE ASS'N OF AM., http://www.mpaa.org/contentprotection (last visited May 13, 2011).

287 JESSICA LITMAN, DIGITAL COPYRIGHT (2001).

288 The reverse engineering exemption, 17 U.S.C. § 1201(f), was held inapplicable in three cases: *Davidson & Associates v. Jung*, 422 F.3d 630 (8th Cir. 2005); *Sony Computer Entertainment America, Inc. v. Divineo, Inc.*, 457 F. Supp. 2d 957 (N.D. Cal. 2006); *Universal City Studios v. Reimerdes*, 111 F. Supp. 2d 294 (S.D.N.Y. 2000), *aff'd sub nom. Universal City Studios, Inc. v. Corley*, 273 F.3d 429 (2d Cir. 2001). It was treated favorably in dicta in one case, *Lexmark International, Inc. v. Static Control Components, Inc.*, 387 F.3d 522 (6th Cir. 2004). The encryption research exemption was held inapplicable in *Reimerdes*.

the scope of protection.[289] Exceptions such as fair use in copyright law have a rich precedential history that could guide judges in tailoring protection appropriately.[290]

Doctrine-specific rules also suffer drawbacks, though. First, protection for software research would need to measure eligibility in a purposive fashion rather than based on formal characteristics or descriptions of activity. Black hat and white hat hackers perform the same type of research; until they disseminate their findings, only their goals differ.[291] If it is not clear what a researcher plans to do with vulnerability information, a court may be risk-averse and block dissemination.[292] Second, judges—especially those unfamiliar with computer technology—may be skeptical of the value of independent security research (rather than that conducted by a vendor), and will likely suffer from information asymmetry, particularly when confronted with ex parte requests for temporary injunctive relief. A vendor's portrayal of the risks from a rogue teenage hacker may swamp calculations of the greater public interest in salience.[293] Finally, vendors would likely engage in strategic behavior. Many complaints against researchers allege multiple violations from different IP doctrines: Mike Lynn faced claims for copyright infringement and misappropriation of trade secrets; Billy Hoffman and Virgil Griffith were accused of trade secret and trademark violations.[294] If protections for security research varied by doctrine, software firms would simply recast their claims against hackers in the relevant theory with the least protection. A shield with holes is nearly as ineffective as no shield at all.

Overall, a statute specifically protecting software security research comports best with this Article's goals. It would focus on the activity to be protected, and not on the form in which a vendor's threat appears. Similarly, a shield law would protect researchers across jurisdictions, as well as across areas of IP. Lastly, a specific statute may be cost saving: it guides courts on

---

289 *See generally* GUIDO CALABRESI, A COMMON LAW FOR THE AGE OF STATUTES (1982); ROSCOE POUND, THE SPIRIT OF THE COMMON LAW (1921).

290 *See* Pierre N. Leval, *Toward a Fair Use Standard*, 103 HARV. L. REV. 1105 (1990).

291 *See* Bryan Smith et al., *Ethical Hacking: The Security Justification, in* ETHICS AND ELECTRONIC INFORMATION 148 (Barbara Rockenbach & Tom Mendina eds., 2003).

292 Daniel Kahneman et al., *Anomalies: The Endowment Effect, Loss Aversion, and Status Quo Bias*, 5 J. ECON. PERSP. 193, 199–203 (1991).

293 *See generally* Derek E. Bambauer, *Shopping Badly: Cognitive Biases, Communications, and the Fallacy of the Marketplace of Ideas*, 77 U. COLO. L. REV. 649, 692–94 (2006); THALER & SUNSTEIN, *supra* note 65, at 24–26, 33–34.

294 Verified Complaint, *supra* note 157, at 6–7.

how to evaluate researchers' actions, and notifies researchers about how to avoid liability. While a shield law faces political challenges, we prefer it as a more effective solution.

C. Changing the Hacker Image

> *It's so easy to impress judges with heavily connoted words like "virus", "pirate", "terrorist", "hacker", and it's so difficult on the other hand to explain the scientific method and the deep curiosity that makes us analyze how software works and find their flaws.*[295]

The term *hacker* is a loaded one. It connotes not only technical skill, but also a disregard for rules and, at times, a malicious enjoyment in finding flaws and wreaking havoc.[296] Researchers like being seen as outlaws rather than nerds. However, these normative associations have real drawbacks along with psychological benefits.[297] Judges, journalists, and the general public may perceive a "hacker" as inherently threatening, and react accordingly.[298] We propose that the research community attempt to mitigate this semantic problem.

Hackers suffer an inherent disadvantage in how others are likely to perceive their work, in three ways. First, an aggrieved software vendor possesses first-mover advantage: the firm is generally the party that frames the dispute for a court by filing a complaint or a request for temporary injunctive relief, which often occurs ex parte. Second, plaintiffs may mix allegations of criminal liability with IP claims, portraying the hackers as vandals or thieves.[299] Finally, *intellectual property* itself is a normatively loaded term that confers an advantage on software providers: researchers are seen as meddling, interfering with, or damaging someone else's valuable possession.[300]

While the first two of these challenges are hard to remedy, researchers can shift the rhetorical debate surrounding their use of others' IP. Even real

---

[295] Rik Lambers, *Guillermito: Reverse Engineering & Scientific Research*, CoCo (Jan. 11, 2005), http://constitutionalcode.blogspot.com/2005/01/guillermito-reverse-engineering.html (quoting Guillame Tena's discussion of his trial for copyright infringement in France for publishing research on flaws in Tegam's Viguard antivirus software; Tegam labeled Tena a "terrorist").

[296] *See* GRAHAM, *supra* note 26, at 51.

[297] *See generally* GEORGE LAKOFF & MARK JOHNSON, METAPHORS WE LIVE BY (1980) (describing the power of linguistic framing to affect cognitive perception).

[298] *See* DOUGLAS THOMAS, HACKER CULTURE 5 (2002) ("This is the common perception of today's hacker—a wily computer criminal . . . .").

[299] *See, e.g.*, *supra* notes 163, 202, 213, and accompanying text.

[300] *See* Fisher, *supra* note 285, at 84–86.

property doctrine permits unauthorized use when there is a compelling reason,[301] such as: necessity[302] or emergency,[303] countervailing social need (such as access to social services),[304] or even customary practice.[305] And intellectual property is most commonly viewed in the United States as a utilitarian bargain between creators and the public, where the state confers limited monopoly rights to attain social benefits, such as information production and distribution.[306] Those rights are circumscribed by exceptions, such as the nominative use doctrine[307] or privileges for socially beneficial actors, such as public libraries,[308] that safeguard valuable though nonpermissive uses.[309] Researchers should therefore emphasize not the potential harm to software companies from vulnerabilities, but the benefits to consumers from fixing those bugs (or, put another way, the risks to consumers if a vendor fails to do so). Property law operates most strongly to limit owners' rights when there are significant externalities involved. Software is a canonical example. To use a real property analogy: researchers should emphasize the interests of the tenants (users) to counteract the claims of the owners. By portraying their work as aligned with users' needs, hackers can mitigate the power of the property analogy employed by software vendors.

Shifting perceptions is difficult. If security researchers want to alter their public image, two strategies are possible. First, they could seek to reclaim the term *hacker*. Initially, a hacker was someone who probed or modified hardware or software to see how it worked, and perhaps to change its function.[310] However, the term increasingly connotes one whose activities are illegal, and perhaps malicious (though discerning researchers refer to the latter as "crackers").[311] To return *hacker* to its lexical roots requires three things, in

---

301 *See generally* Gregory S. Alexander, *The Social-Obligation Norm in American Property Law*, 94 CORNELL L. REV. 745 (2009); Larissa Katz, *Exclusion and Exclusivity in Property Law*, 58 U. TORONTO L.J. 275 (2008).

302 *See, e.g.*, Vincent v. Lake Erie Transp. Co., 124 N.W. 221, 222 (Minn. 1910).

303 *See, e.g.*, Ploof v. Putnam, 71 A. 188, 189 (Vt. 1908).

304 *See, e.g.*, State v. Schmid, 423 A.2d 615 (N.J. 1980); State v. Shack, 277 A.2d 369, 373 (N.J. 1971).

305 *See, e.g.*, M'Conico v. Singleton, 9 S.C.L. (2 Mill) 244, 246 (1818) (recognizing right of hunters to enter unenclosed rural land).

306 *See generally* Stewart E. Sterk, *Rhetoric and Reality in Copyright Law*, 94 MICH. L. REV. 1197, 1204–26 (1996).

307 *See, e.g.*, New Kids on the Block v. News Am. Publ'g, Inc., 971 F.2d 302, 307–08 (9th Cir. 1992) (discussing nominative use).

308 17 U.S.C. § 108 (2006).

309 *See, e.g.*, *id.* §§ 110, 117.

310 *See, e.g.*, RAYMOND, *supra* note 12, at 5–26.

311 *See id.* at 231–50.

ascending order of difficulty: finding a new term for those who invade systems or crack software with ill intent; convincing the security research community to adopt the new term and employ it with some consistency; and convincing others (particularly the media) to follow the new usage pattern. Alternative terms, such as *cracker*, are readily available. However, the research community seems unwilling to shift usage—in part because some like the outlaw image that *hacker* currently provides. Even if vulnerability researchers take up new terms, it is not clear that such a change will spread to the wider public, particularly since *hacker* is evocative.

The second option is for legitimate researchers to abandon *hacker* to the black hats. One way to do this would be to embed the term *hacker* in federal criminal law, such as by defining it in the Computer Fraud and Abuse Act.[312] This would formalize the equivalence between hackers and black hats. Legitimate researchers would employ a new term to describe those who conform to laws and norms governing software security, and would insist (to the degree they are able) that others refer to them by that moniker. While the adoption challenges described above remain, a new term will lack the cognitive inertia that *hacker* possesses, which may mitigate these issues. Moreover, researchers can seek to shift the analogy that dominates vulnerability analysis. If they portray their role as similar to whistleblowers,[313] independent testing companies such as Consumers Union,[314] or watchdogs such as the Center for Science in the Public Interest,[315] their work is more likely to be treated as legitimate. The term needs to be pithy, appealing, and different from *hacker*; we offer *bug hunter*, *cyber-watchdog*, and *security researcher* as possibilities, but hope others will introduce more catchy options.[316] This shift in perception—drawing a distinct rhetorical line between white hat and black hat researchers—will benefit researchers. As researchers'

---

312 We thank Scott Velez for this insight.

313 *See, e.g.*, JOHN F. KELLY & PHILLIP K. WEARNE, TAINTING EVIDENCE 3 (1998) (describing role of Frederic Whitehurst in revealing wrongdoing at the Federal Bureau of Investigation's crime laboratory).

314 *See, e.g.*, *Cutting Surgical Infections*, SAFE PATIENT PROJECT (Apr. 23, 2009), http://www.safepatientproject.org/cutting_surgical_infection.pdf (documenting hospitals' rates of infections during surgery).

315 *See, e.g.*, *Bayer Ads Misleading Men About Prostate Cancer, Says CSPI*, CTR. FOR SCI. IN THE PUB. INTEREST (June 18, 2009), http://www.cspinet.org/new/200906181.html (summarizing a watchdog group's action to stop Bayer Healthcare's false advertising).

316 We recognize our lack of skill in developing pithy terminology, and thank Shubha Ghosh (*paladins*), Zahr Said (*breakers*), and Adam Candeub (*vanguards*) for excellent suggestions for the new moniker. Dan Guido informed us that some researchers have adopted the term *busticati* for this purpose.

work appears less threatening, legal measures to restrict its production and dissemination will appear less necessary.

D. Freeing Markets

The market for information about software vulnerabilities is not a well-working one due to high transaction costs, information asymmetry, the risks of strategic behavior, and time pressure. We propose two changes that will ameliorate these issues.

Both researchers and vendors are reluctant participants in transactions involving vulnerability data, in part because of transaction costs. A hacker who discovers a flaw must determine which party to attempt to do business with (for example, deciding between the vendor, the vendor's customers, or security consulting firms) and then who to contact (for example, a development team, legal counsel, or management). For their part, vendors must separate legitimate inquiries from attempts at fraud or blackmail. They also must assess whether a vulnerability is a known problem, whether the researcher or others have working code to exploit it, and whether the seller is sufficiently trustworthy to enter into a transaction.

Even once a willing seller locates, and communicates with, a willing buyer, the parties will have difficulty coming to terms due to information asymmetry.[317] There is no price list, or set of criteria, to determine what a bug is worth.[318] Unsurprisingly, vendors tend to value vulnerability information less than researchers do. The market for security flaws is an illiquid one, since transactions are sporadic and often secret.[319] The lack of reference data for pricing means that vendors and researchers may fail to strike deals that would benefit both parties, since they may err (or simply differ) in assessing the data's value. Moreover, reducing information asymmetry through sharing is challenging.[320] For sellers, presenting their wares to software companies is chancy, not merely because doing so may put them at legal risk, but because

---

317 *See* Kenneth J. Arrow, *Economic Welfare and the Allocation of Resources for Invention*, *in* THE RATE AND DIRECTION OF INVENTIVE ACTIVITY 609, 616 (Nat'l Bureau of Econ. Research, Special Conference Ser. 13, 1962).

318 *See* Greenberg, *supra* note 101 (describing a researcher's difficulty in valuing a server vulnerability, even with a willing buyer).

319 *See generally* Karthik Kannan & Rahul Telang, *Market for Software Vulnerabilities? Think Again*, 51 MGMT. SCI. 726 (2005) (describing the inefficiencies of a market-based vulnerability-disclosure system).

320 *See generally* E. Allan Farnsworth, *Precontractual Liability and Preliminary Agreements: Fair Dealing and Failed Negotiations*, 87 COLUM. L. REV. 217, 267 (1987).

sharing findings could destroy their value. This results from Arrow's paradox: it is hard to demonstrate the value of a security flaw without revealing information sufficient to permit the vendor to remedy it.[321]

Finally, the risk of strategic behavior weakens prospects for a successful bargain. Researchers worry about misappropriation. Arrow's paradox presents a hard choice: disclose too little, and vendors may not believe the problem is real; disclose too much, and a software company may take the information without compensation. Vendors, in turn, have trouble guaranteeing that a researcher who shares data with them is not also sharing it on the black market. Paying hackers for bugs may also tempt researchers to target a company's software. Fears about the other party's behavior effectively decrease the value of a deal for both sides (due to increased risk that the bargain will unravel) and may lead to additional costs from preventive measures.

Lastly, time pressures shrink the window for vendors and researchers to reach an agreement. Hackers correctly perceive that their vulnerability information has a limited viable lifetime.[322] Other researchers may discover the same weakness and either launch an attack or offer a competing bargain. The vendor may change its code, deliberately or inadvertently remedying the problem. If the hacker has revealed any information to the software company, the firm may be able to reverse engineer the vulnerability from that limited data, making the research worthless. Firms, too, face time constraints. If one person has found a weakness in their code, others are likely to do so as well. A researcher frustrated by the pace of negotiations may turn to the black market. Finally, vendors require lead time to write, compile, test, and distribute patches. To address a bug, a vendor needs as much time as possible to generate a fix and to get customers to install it.

There are significant structural barriers to market transactions between information suppliers (researchers) and consumers (vendors). To reduce these impediments, we propose that a trusted third party act as a voluntary coordinator or clearinghouse for vulnerability deals. We envision this intermediary playing three roles. First, it would archive and validate vulnerability data for researchers. This would allow a hacker to claim credit

---

321 Arrow, *supra* note 317, at 616.

322 Stefan Frei et al., *Modelling the Security Ecosystem—The Dynamics of (In)Security*, 8 WORKSHOP ON ECON. INFO. SECURITY 3–7 (2009), *available at* http://www.techzoom.net/papers/weis_security_ecosystem_2009.pdf.

for discovering a bug, and to store diagnostic data and any exploit code. Registries with trusted third parties have been used successfully to overcome the challenges of Arrow's paradox in other contexts, such as unsolicited manuscripts for television shows and movies.[323] Second, it would maintain contact information for vendors and researchers. The third party might offer anonymous referrals, where the identity of a hacker or vendor is known to the coordinator but not to the other party. This could encourage researchers who are risk-averse to share discoveries through legitimate channels. Finally, the trusted third party could play a reputational role. It could make available data about previous reports and transactions, perhaps in summary form, to help vendors and researchers establish trustworthiness.[324] A more interventionist role might have the intermediary act as a third-party beneficiary to a nondisclosure agreement between a seller and buyer, allowing the coordinator to enforce bargains and to police adherence through both legal means and reputational sanctions (such as disclosing violations publicly).

Existing security organizations, such as Computer Emergency Response Team's Coordination Center (CERT/CC) at Carnegie Mellon University's Software Engineering Institute, are well positioned to act as clearinghouses.[325] CERT/CC has a strong reputation in the computer security field and acts as a key channel of information distribution about vulnerabilities. CERT Coordination Center partners with both private- and public-sector entities in the field, and its Knowledgebase already contains data on thousands of reported vulnerabilities.[326] Other entities, such as the Internet Storm Center,[327] MITRE,[328] and perhaps even government-sponsored organizations such as the National Vulnerability Database,[329] might also act as intermediaries. The critical issue for the coordinating entity is credibility: it must be trusted, both

---

[323] *See* Robert P. Merges, *Contracting into Liability Rules: Intellectual Property Rights and Collective Rights Organizations*, 84 CALIF. L. REV. 1293, 1366–68 (1996) (describing script registry operated by Writer's Guild of America); WGAWREGISTRY.ORG, http://www.wgawregistry.org/webrss (last visited May 13, 2011).

[324] *Cf.* Jonathan Zittrain, *Privacy 2.0*, 2008 U. CHI. LEGAL F. 65, 93–97 (describing reputation rating systems).

[325] *CERT Coordination Center (CERT/CC)*, COMPUTER EMERGENCY RESPONSE TEAM, http://www.cert.org/certcc.html (last visited May 13, 2011).

[326] *CERT Knowledgebase*, COMPUTER EMERGENCY RESPONSE TEAM, http://www.cert.org/kb (last updated Apr. 18, 2008).

[327] *About the Internet Storm Center*, SANS INTERNET STORM CTR., http://isc.sans.org/about.html (last visited May 13, 2011).

[328] COMMON WEAKNESS ENUMERATION, http://cwe.mitre.org (last updated May 11, 2011).

[329] NAT'L VULNERABILITY DATABASE, http://nvd.nist.gov (last updated May 13, 2011).

by vendors and by researchers.[330] To this end, it might be necessary to insulate the intermediary itself from liability based on holding bug data or interacting with its constituents. The clearinghouse could likely achieve immunity through private bargains—vendors and researchers could be required to waive claims against it as a condition of participation—but if necessary, the "shield law" discussed above should include such protections.

A benefit of this approach is that both vendors and researchers would likely utilize a trusted third party system voluntarily, because it reduces their costs and risks. For example, the combination of the registry and the reputational metadata would help vendors decide which researchers are worth the cost of entering into a nondisclosure agreement to further inspect a claimed vulnerability. This increases the likelihood that this part of the Hacker's Aegis would be adopted, and used.

Use of a voluntary coordinating intermediary would help reduce costs that researchers and vendors alike face in exchanging information about vulnerabilities, making legitimate transactions easier and more likely.

E. Challenges

There are at least two potential challenges that our proposed reforms might confront. The first is that our legal reforms might be underinclusive. The second is the risk of strategic behavior based on the legal safe harbor we propose. In this section, we address each issue.

Our legal proposal contemplates a shield from civil liability under intellectual property claims or causes of action. However, it does not encompass other theories of liability—in particular, tort claims, civil claims under the Computer Fraud and Abuse Act, and criminal prosecution. We have four reasons for crafting the safe harbor to leave these legal tools available. First, in analyzing legal threats against security researchers, IP claims predominate. We view establishing a shield against them as a first step, but not necessarily a final one. If software companies shift to using alternative theories such as tort and CFAA claims, with similar chilling effects, we would advocate expanding the safe harbor to exclude such theories.

---

[330] This could weigh against a government entity acting as intermediary, as security researchers may be reluctant to reveal information to a sovereign with the power to prosecute them.

Second, tort claims are typically weaker than IP ones. Common law doctrines such as trespass to chattels have largely been displaced by software- and internet-specific statutes.[331] Successful suits under trespass to chattels are relatively rare, and the theory has been questioned by leading courts such as the California Supreme Court.[332] Other tort claims, such as tortious interference with business expectations or prospective economic advantage, typically recapitulate IP claims in slightly different form.[333] In addition, proof of actual damage is required for tort claims, such as interference with prospective advantage, and injunctive relief is atypical.[334] These factors reduce the risk from tort theories to software researchers.

Third, the Computer Fraud and Abuse Act contains a built-in limitation on civil liability that offers protection to security researchers. To maintain a cause of action under the CFAA, a plaintiff must demonstrate economic damages of at least $5,000 in a one-year period, impairment of a person's medical treatment, physical injury to a person, threat to public health or safety, or damage to a U.S. government computer.[335] A researcher testing software on a computer under her control is unlikely to contravene any of these limits, reducing the threat of the CFAA.

Finally, our proposed changes leave security researchers vulnerable to criminal charges, under the Computer Fraud and Abuse Act and other statutes.[336] This is deliberate. We believe that criminal sanctions are necessary to deter strategic behavior by black hat hackers, who may try to fit their activities within the contours of the safe harbor.[337] Criminal law acts to reinforce any gaps within the liability shield.[338] If a hacker's activity, though

---

[331] *Compare* CompuServe Inc. v. Cyber Promotions, Inc., 962 F. Supp. 1015 (S.D. Ohio 1997) (issuing injunction against spammer based on trespass to chattels theory), *with* Facebook, Inc. v. Guerbuez, No. C08-03889-JF-HRL, 2008 U.S. Dist. LEXIS 108921 (N.D. Cal. Nov. 21, 2008) (issuing judgment in similar case against spammer based on CAN-SPAM statute).

[332] *See, e.g.*, Intel Corp. v. Hamidi, 71 P.3d 296 (Cal. 2003).

[333] *See, e.g.*, Robi v. Five Platters, Inc., 838 F.2d 318, 323–24 (9th Cir. 1988) (finding claim preclusion of trademark claim based on prior decision regarding interference with contractual relations).

[334] *See, e.g.*, Durasys, Inc. v. Leyba, 992 F.2d 1465 (7th Cir. 1993).

[335] 18 U.S.C. § 1030(g) (2006 & Supp. II 2008).

[336] *See, e.g.*, *id.* §§ 1030, 2511.

[337] *But see, e.g.*, Mark Rasch, *German Hacker-Tool Law Snares . . . No-One*, REGISTER (June 7, 2009, 08:02 AM), http://www.theregister.co.uk/2009/06/07/germany_hacker_tool_law (arguing German cybercrime law resulted in security companies leaving that country, despite lack of prosecutions under it).

[338] *Cf.* Miriam H. Baer, *Linkage and the Deterrence of Corporate Fraud*, 94 VA. L. REV. 1295 (2008) (analyzing the effectiveness of neoclassical and alternative deterrence theories in the context of criminal, corporate anti-fraud laws—an area in which perpetrators' conduct is more likely to be discovered if they discontinue the fraud).

protected from civil claims, causes sufficient harm, a software company is likely to be able to convince a prosecutor to file charges. The risk of overdeterrence from criminal law remains, but it is likely no worse than under current circumstances. In addition, prosecutors are more likely to arrive at an objective assessment of whether a hacker's behavior is beneficial or malicious than the vendor whose software has been targeted. Thus, retaining criminal liability for hacking serves a helpful deterrence function, and should not create additional chilling effects for security researchers.

There is one aspect of CFAA criminal liability that may give hackers pause and thereby create overdeterrence: the ban on damage that affects ten or more computers.[339] This provision, adopted to deal with threats from viruses and worms, may be problematic when researchers probe cloud computing. Services such as Gmail run on multiple servers, and their storage units (such as storage array networks (SANs) or network-attached storage (NAS)) may comprise computers in their own right.[340] If criminal CFAA liability interferes with testing of cloud-computing security, we propose two fixes. First, the legal safe harbor could modify the CFAA to condition liability on accessing a larger number of computers—perhaps 100. This would maintain liability for virus creators and distributors, but would reduce the threat to researchers. Second—and likely more promising—cloud-computing services should be treated as a single computer under the CFAA. The appeal of cloud computing is that it appears to users as a single service or computer. Moreover, the number of computers accessed during a cloud-computing session is under the control of the service provider. This creates a risk of strategic behavior: providers could ensure that any transaction affected ten or more computers, creating the possibility that *any* claimed damage would generate potential criminal liability. However, we believe such alteration should be withheld until there is more evidence of harm to cloud computing security research. As described above, treating cloud computing as a single computer would effectively remove part of the CFAA as a resource for providers, which is why we believe such a change should wait for more evidence of a problem.

This Article's proposals are a first step toward mitigating IP law's unhelpful channeling effects for software security research. Its proposed legal

---

339 18 U.S.C. § 1030(c)(4)(A)(i)(VI) (2006); *id.* § 1030(g) (2006 & Supp. II 2008).

340 *Cf.* SUN MICROSYSTEMS, INTRODUCTION TO CLOUD COMPUTING ARCHITECTURE (2009), *available at* http://webobjects.cdw.com/webobjects/media/pdf/Sun_CloudComputing.pdf (discussing the transformative nature of cloud computing and outlining considerations that businesses should take when implementing such a system).

reforms do not alter tort theories and civil CFAA claims regarding hacking because their built-in doctrinal safeguards should be sufficient. Should this prove incorrect, we propose revisiting the scope of the legal shield to address such risks. And, our proposal retains criminal liability as a necessary deterrent to counter strategic behavior by malicious hackers.

CONCLUSION

This Article argues that intellectual property law impedes the dissemination of socially valuable research into software security flaws. By reducing the threat of civil IP liability for researchers, the cost of legitimate transactions, and the specter of harm from the term *hacker*, our proposed reforms would improve software security and decrease users' risks.

The paper's conclusions have repercussions beyond software—research into security occurs in the physical world as well as the digital one.[341] A graduate student in computer science created a web application that generates a boarding pass sufficiently realistic to deceive Transportation Security Administration screeners.[342] For his efforts to show the ineffectiveness of airport security, he had his computer seized, was questioned by the FBI, and was denounced by a U.S. Congressman.[343] Cyclists who relied on Kryptonite bike locks were startled when security consultant Chris Brennan showed how to open the locks using a plastic pen.[344] Medeco locks—considered so secure that they are used at the White House and the Pentagon—have been hacked using credit cards and sharp scissors.[345] Security researcher Chris Soghoian

---

[341] *Cf.* Peter P. Swire, *A Model for When Disclosure Helps Security: What Is Different About Computer and Network Security?*, 3 J. ON TELECOMM. & HIGH TECH. L. 163 (2004) (contrasting free disclosure versus secrecy and analyzing the ability of each to aid or diminish security in traditional contexts or in computer network contexts).

[342] CHRIS'S NWA BOARDING PASS GENERATOR, http://www.dubfire.net/boarding_pass (last visited May 13, 2011); *see also* Christopher Soghoian, Insecure Flight: Broken Boarding Passes and Ineffective Terrorist Watch Lists (July 19, 2007) (unpublished manuscript), *available at* http://papers.ssrn.com/sol3/papers.cfm?abstract_id=1001675.

[343] Robert Lemos, *FBI Raids Home of Boarding-Pass Creator*, SECURITYFOCUS (Oct. 30, 2006), http://www.securityfocus.com/brief/342; Jonathan Silverstein, *Web Site Lets Anyone Create Fake Boarding Passes*, ABC NEWS (Oct. 27, 2006), http://abcnews.go.com/Technology/story?id=2611432&page=1; Ryan Singel, *Congressman Ed Markey Wants Security Researcher Arrested*, WIRED THREAT LEVEL (Oct. 27, 2006, 10:57 AM), http://www.wired.com/threatlevel/2006/10/congressman_ed_/.

[344] Leander Kahney, *Twist a Pen, Open a Lock*, WIRED (Sept. 17, 2004), http://www.wired.com/culture/lifestyle/news/2004/09/64987; Filouphil, *Kryptonite Bikeforum*, YOUTUBE (Jan. 30, 2007), http://www.youtube.com/watch?v=-9bN0zfMFW4 (showing how to pick the lock in seconds).

[345] Kim Zetter, *Researchers Crack Medeco High-Security Locks with Plastic Keys*, WIRED THREAT LEVEL (Aug. 8, 2008, 11:19 AM), http://www.wired.com/threatlevel/2008/08/medeco-locks-cr.

published a guide to loopholes and exploits in consumer credit practices that enable attackers to modify their credit reports and obtain loans for which they could not otherwise qualify.[346] "Locksporters" test every lock on the market with tools from custom-made hooks to beer cans.[347] Each of these activities produces valuable information about how safe we really are, and each has been subject to legal threats. It may be necessary to extend the Hacker's Aegis to protect them as well. Bug hunting may simply be one exemplar of a peer-production activity with societal benefits that is impeded by law.

Software security research is helpful, and intellectual property law interferes with it. This Article explains how to shield the white hats from the gray suits.

---

[346] Christopher Soghoian, *Manipulation and Abuse of the Consumer Credit Reporting Agencies*, FIRST MONDAY (Aug. 3, 2009), http://www.uic.edu/htbin/cgiwrap/bin/ojs/index.php/fm/article/view/2583/2246.

[347] Charles Graeber, *The Lock Busters*, WIRED (Feb. 2005), http://www.wired.com/wired/archive/13.02/lockbusters.html; Trine Tsouderos, *Pick a Lock. For fun. (It's Legal Too)*, CHI. TRIBUNE, Sept. 25, 2008, at C1.