CSI Working Group on Web Security Research Law

Inaugural Report, June 11, 2007
Computer Security Institute

By **Sara Peters**, Editor, Computer Security Institute

Contributing Experts:

Brian Chess
founder and chief scientist, Fortify Software

Jennifer Granick
executive director, Center for Internet and Society, Stanford Law School

Jeremiah Grossman
chief technology officer, WhiteHat Security

Billy Hoffman
lead researcher, SPI Labs

John Lynch
deputy chief, Computer Crime and Intellectual Property Section, Criminal Division, U.S. Department of Justice

Scott Parcel
vice president of engineering, Cenzic

Jonathan Rusch
special counsel for fraud prevention, Criminal Division, U.S. Department of Justice

Lee Tien
senior staff attorney, Electronic Frontier Foundation

Jacob West
manager, Security Research Group, Fortify Software

Web vulnerability research is subject to vastly different legal restrictions than software vulnerability research. As Web technology advances, as more sensitive activities occur online, as more Web sites dress themselves up in Web 2.0 apparel, any impediments to Web security research must be reexamined.

The Computer Security Institute's Working Group on Web Security Research Law was created in April 2007 to start a dialogue between Web researchers, computer crime law experts and law enforcement officials on these issues, and to advance the collective understanding of the challenges facing all parties. Though consensus within the task force is noted when it exists, the purpose is not to espouse any particular position. Rather, the goals are to identify all the legal, ethical, social and technological considerations feeding into this issue; to provide detailed background information about Web research methods and legal precedents; and to explain all the arguments on each question.

Over the course of two months, working group members tackled an ever-widening agenda, communicating and collaborating through conference calls and e-mails. This report is the sum—for the moment—of the group's exploration of Web security research law.

The working group conversed with Daniel Cuthbert and Eric McCarty (discussed below), but neither directly contributed to this report.

Introduction

The next generation of Internet applications, Web 2.0, makes it possible for people with no knowledge of Web development to conduct many activities we never knew we wanted to do online—pay bills, order medication, trade photos of our children with everyone we know. (If you find you want to donate money to the World Wildlife Foundation, buy a stuffed panda for your niece, sign up for an RSS feed of panda-related news, e-mail a panda trivia game to a friend, piece together a panda puzzle, while watching real-time streaming video from the Panda Cam, you can even do that.[1])

As technology advances, as Web applications become more ubiquitous, as more confidential data is transmitted over the Internet, as more sensitive data is stored on Web servers, the challenge of securing Web 2.0 becomes more urgent.

However, unlike typical software developers, Web developers do not benefit from (nor chafe at) the security community's detection and disclosure of vulnerabilities within their applications. The law makes it so.

Security researchers are legally permitted to publicly disclose *software* vulnerabilities with near abandon. H.D. Moore can tear into browser software and run a Month of Browser Bugs, Kevin Finisterre and LMH can slice apart operating systems and run a Month of Apple Bugs and David Litchfield can expose database bugs and rage against Oracle's long times-to-patch. Software vulnerability researchers have a wide selection of open-source vulnerability scanners to choose from, and they can write proof-of-concept exploits and post them on their blogs.

Conversely, if a Web security researcher has not been contracted by a Web site owner to find weak spots, he could, under some laws, find himself in prison for simply looking for a Web site vulnerability, much less disclosing it publicly.

The rationale for the difference is that when H.D. Moore or Kevin Finisterre rip apart operating systems and browser apps they do so on their own machines. They legally have the right to experiment upon—even completely destroy—their own computer systems. Conversely, the Web researcher, in most cases, must perform his activity on a server owned by someone else. Thus any damage that results from this activity is suffered by this other party.

It is true that software security researchers can get tangled in legal snares if their research methods brazenly defy copyright law or the software vendor's end-user licensing agreement. Yet those are not criminal offenses. Though hefty fines and damaged reputations are distinct possibilities, the researchers will not go to jail for these actions. A background check will not list a felony on their record.

Within the information technology community, heated debates ensue over the issue of "responsible disclosure" of software vulnerabilities, and what in fact the term means.

Many software vendors strongly urge vulnerability researchers to report their findings to the vendor, and only the vendor, thereby giving the vendor time to patch the hole before malicious hackers learn of the flaw and exploit it. The other pole espouses "full disclosure," the idea that the vendor community cannot be relied upon to secure its software, and thus researchers—to give the public an opportunity to protect themselves—should immediately inform the general public when vulnerabilities are found.

Advocates of this policy also assert that full disclosure puts pressure on the software vendors to take security more seriously, and shorten times-to-patch. The vendor community counters that argument with the point that their patches must undergo extensive testing to properly consider all their customers' unique configurations and avoid interoperability troubles.

In between those poles lies a spectrum of opinions, and perhaps the most common definition of "responsible disclosure"—that software vendors should be the first to know of vulnerabilities and given a reasonable time to patch them; and if the vendor does not adequately respond in this time period, the public should then be notified. Opinions on what qualifies as a "reasonable" time period vary greatly.

Researchers are often urged to avoid giving away too much, discouraged from providing complete exploits. Other researchers suggest that the same skills required to find vulnerabilities can be used to write patches, and thus vulnerability researchers should provide work-arounds and third-party patches simultaneously with the disclosure [1]. Other third-party entities have arisen with the mission to patch critical "zero-day" flaws as temporary fixes to tide users over until an official vendor-released patch is available [2].

In essence the responsible disclosure issue—as it pertains to software vulnerability research—is an *ethical* debate. Yet ethics do not rule Web vulnerability research and disclosure; rather Federal and state laws do. [2]

[1] http://animal.discovery.com

[2] More accurately, laws do not exactly prohibit Web vulnerability *disclosure*. The crime, rather is in the vulnerability *research*. Of course, to disclose a vulnerability, a researcher must first find it–and it is the finding that can get them into trouble.

Is the legal distinction between software vulnerability research/ disclosure and Web vulnerability research/ disclosure sensible?

Do these laws unduly restrict the work of Good Samaritan Web security researchers?

Should Web site owners be held liable for security weaknesses, or should the researchers who find them?

Considering how many critical activities now take place over the Internet, and how much sensitive information travels over the Net, does the legal climate ultimately degrade the security of the Internet as a whole?

Should the law be changed?

There is by no means a consensus opinion.

This working group was created to investigate these questions. By starting a dialogue between Web researchers, computer crime law experts and law enforcement officials, we hope to advance the collective understanding of the challenges facing all parties. Though consensus within the working group is noted when it exists, the purpose is not to espouse any particular position. Rather, the goals are to identify all the legal, ethical, social and technological considerations feeding into this issue; to provide detailed background information on Web security research techniques and legal precedents; and to explain all the arguments on each question.

It seems a sensible place to begin discussion is with relevant case studies, citing significant legal strictures.

CASE STUDIES

CASE STUDY 1: DANIEL CUTHBERT

Defendant:

Daniel Cuthbert; 28 (at time of arrest); citizen of the United Kingdom; professional security contractor for a central London bank; no prior convictions [3]

Charges:

❏ credit card fraud

Under the Computer Misuse Act 1990 (CMA), United Kingdom:

❏ “unauthorized access to computer materials” [4]
  ✧ maximum sentence: six months imprisonment

❏ “unauthorized modification of computer material”
  ✧ maximum sentence: five years imprisonment [5]

Tried:

Oct. 5, 2006–Oct. 6, 2006, Horseferry Magistrate’s Court, Westminster, London [3]

Decision:

❏ “unauthorized access to computer materials = guilty

❏ “unauthorized modification of computer material = not guilty

❏ credit card fraud = charges dropped [3]

Sentence:

£400 (700 USD) in fines and £600 (1,050 USD) in court fees [3]

Details of case:

The charges were brought against Cuthbert for attempting to hack into the Disasters Emergency Committee’s (DEC) Web site (www.dec.org.uk). After donating £30 (and an array of personal information) to DEC’s tsunami relief fund, Cuthbert grew suspicious that he’d happened upon a phishing site; he received no confirmation message, the page didn’t reload, and the whole site suffered from what he calls “poor coding.” Finding no way to contact the site administrator, Cuthbert probed a site application with a trivial shell command to test its security; this would later earn him a conviction for unauthorized access to computer material and a charge of unauthorized modification of computer material (for altering the site’s log files). [6]

According to Cuthbert, the story begins at about 3 p.m. on New Year’s Eve, 2004. He was online, sitting at his machine, when he spotted a link to a tsunami relief fund accepting donations. He clicked on the link, arrived at DEC’s Web site, and donated £30. [6]

“I gave them my name, address, credit card information; all the things I hate giving out,” he said. “I clicked ‘submit’ and… nothing happened. There was no confirmation. The page didn’t even reload. Nothing. So, alarm bells start going off.” [6]

Cuthbert suspected a phishing attack and ran quick tests to see if the site had any security, assuming that a here-today-gone-tomorrow phishing site wouldn’t waste time on security. He said he had no idea that he was breaking a law. After the tests, he decided the site was legitimate (if not terribly good) and forgot all about it. [6]

But the site’s intrusion detection system (IDS) did not forget. The secure donation site was hosted by British Telecom (BT), whose IDS red-flagged the incident. BT then reported it to the London Metropolitan Police Service, thus launching the investigation. [3]

Jan. 10 eight police officers (according to Cuthbert) showed up at Cuthbert's place of business and arrested him for unauthorized access to computer material, unauthorized modification to computer material and credit card fraud. The fraud charges were later dropped—it seems Cuthbert's card had been reported stolen, so the charge was, in essence, for defrauding himself.

According to Cuthbert, during his arrest the police searched him, administered drug tests and took a sample of his DNA. [6]

The CMA takes a hard line on "hacking"—conviction is not incumbent upon damage being done [4]. However it does require that the defendant know he is causing a computer to perform a function with intent to secure unauthorized access [7].

CASE STUDY 2: U.S. VS. ERIC MCCARTY

Defendant:

Eric McCarty, 25 (at time of arrest), U.S. citizen, independent researcher, no prior convictions

Charges:

Under the Computer Fraud and Abuse Act (U.S. Code 18, Section 1030; 1986, 1996, 2001, 2002)

❏ "knowingly causes the transmission of a program, information, code or command, and, as a result of such conduct, intentionally causes damage without authorization to a protected computer [3] [8]." [9]

  ✧ Maximum sentence: fine up to $250,000 or twice the gross gain or gross loss, whichever is greatest, imprisonment of five years, or both [10]

Decision:

McCarty pled guilty and signed a plea agreement, December 2006 [11]

Terms of agreement:

Six months of house arrest and fines of $36, 761.26. [11]

Details of case:

April 18, 2006 Eric McCarty was charged under the U.S. Computer Fraud and Abuse Act, for actions he took June 17, 2005. [12]

In June 2005, McCarty, who did not hold a college degree, wished to apply to the University of Southern California (USC). Before handing over his personal data, McCarty decided to first determine whether or not the online application system was secure enough. [13]

He found that a SQL injection could be performed on the homegrown authentication software, within the online application site, so an attacker could circumvent authentication and access any of the forms in the database. The database contained personal data on 275,000 applicants, dating as far back as 1997. The data included names, addresses, dates-of-birth and Social Security numbers, as well as usernames and passwords used to enter the application site [14]. Learning this, he opted not to apply at that time [4]. [13]

Believing that USC would be unlikely to take a random e-mail from a stranger seriously, McCarty did not report the vulnerability directly to USC. Rather he contacted, under a pseudonym [14], what he considered to be a respected third party, Robert Lemos, a reporter at SecurityFocus. Lemos told USC what McCarty told them. USC never contacted McCarty directly, but rather responded to SecurityFocus, denying the problem. When Lemos relayed that message back, McCarty proved the vulnerability existed by sending Lemos screen shots of seven applicants' records he'd accessed by performing the exploit himself. Lemos forwarded the data to USC. USC then confirmed that the vulnerability did exist. [15]

USC then brought the online application site down for 10 days to make the appropriate fixes. Being that USC is regulated by California data breach notification law [16], the school also mailed notices to all the people in the database whose personal information may have been exposed [5] [17]. The sum cost of these measures was estimated, by USC representatives, at $140,000. [14]

[3] "Protected computer" here means "a computer which is used in interstate or foreign commerce or communication, including a computer located outside the United States that is used in a manner that affects interstate or foreign commerce or communication of the United States." [8]

[4] According to McCarty, he did not apply to the university, but according to the FBI agent's affidavit [14] McCarty was disgruntled because he did apply to the university, and was rejected.

[5] "Personal information" is defined as "the first name or first initial and the last name, plus one of the following: Social Security number; driver's license number; account number, credit or debit card number, in combination with any required security code, access code, or password that would permit access to an individual's financial account." [17]

About a year later, in April 2006, McCarty was charged, and $140,000 in damages was sought. [14] (Further discussion on the definition of "damage" will be discussed below in the sections "Does the law properly assign liability?" and "Should the law change?", subsection "Damage.")

In December 2006 McCarty signed a plea agreement establishing he be on house arrest for six months and pay USC almost $37,000. [11]

McCarty said he had to plead guilty because he simply could not afford the financial expense of going to trial. He believes he could have won the case, if it had. [13]

CASE STUDY 4: NEW ZEALAND POLICE VS. GERASIMOS MACRIDIS

Defendant:

Gerasimos "Gerry" Macridis, age 39 (at time of arrest), citizen of New Zealand, self-employed security consultant, multiple prior convictions for fraud [6]. [18]

Charges:

Under New Zealand Crimes Act 1961, Sec. 252

❏ "accessing of a computer system without authorization"
 ✧ maximum sentence = two years imprisonment [19]

Hearing:

Sep. 27, 2006, Wellington District Court [20]

Decision:

Although Macridis pled guilty to the charge, the case was dismissed by Judge Ian Mill. [20]

Details of case: [7]

The outcome of this case could hardly differ more from the previous two.

May 30, 2006 Macridis, a self-employed security researcher who worked for the New Zealand Government Department of Internal Affairs "on a casual basis," contacted the Reserve Bank of New Zealand. He told bank representatives that he had scanned the computer used to operate the bank's telephone system, and found serious vulnerabilities [8]. [18]

Though there was no contract between the bank and him, Macridis requested payment for this information. He was "refused by Reserve Bank staff who terminated the call." [18]

Macridis then called and e-mailed Telecom Fraud Services, an organization for which he had done consulting work before. He conveyed to Telecom the information about the vulnerability and how to negate it, and again requested payment. [18]

The next day Reserve Bank filed a complaint with the police. According to a summary of facts from a detective constable, "The act of inputting a numerical PIN code for the purpose of accessing the 'Remote Call Divert' (RCD) facility without authority to do so falls under the definition of accessing a computer system without authority, for the purposes of 252 of the Crimes Act 1961." [18]

In September, police executed a search warrant on Macridis' home and seized his computer. In a police interview later that day, Macridis said he had accessed the telephone system in 2004, however "telephone records for May 2006 show he had been accessing the RCD system throughout that time." It is not clear whether "throughout that time" means throughout the month of May 2006, or throughout the period beginning in 2004 and ending in May 2006. [18]

Six days after his arrest Macridis represented himself in court and was granted by the judge his request to discharge the case without conviction. [20]

In his judgment, Judge Mill acknowledged that there were "some very unusual aspects to this case"—not the least of which was the fact that Macridis had been convicted multiple times for fraud, though the last conviction had happened more than ten years prior. [20]

"Mr. Macridis has a history of offending," said Judge Mill, "and there is no doubt that that offending was for dishonesty. The information that has been supplied to me now is that he uses his talents to assist organizations." [20]

Mill noted that the report Macridis wrote was comprehensive and contained critical information. "The information from the bank is that, indeed, this was a serious matter; they viewed it seriously, and for his trouble, [Macridis] is being prosecuted. [20]

[6] The author of the report could not fulfill the necessary requirements under Section 71 (4) of the New Zealand Summary Proceedings Act 1957, to receive documents detailing Macridis' prior convictions. The only information known is that his most recent fraud charge occurred 10 years prior to the charge detailed here.

[7] It should be noted that this is not quite an apples-to-apples comparison, since the defendant accessed telephone systems, not a Web server. Yet, it is closely comparable, since Macridis accessed a system he did not own.

[8] Portions of the summary of facts were suppressed in the version the N.Z. Ministry of Justice sent to the author of the report. The document gives no details on the vulnerabilities themselves.

"He did not use this information for personal gain or to provide access to others, although it is accepted that he expected some payment for it," said Judge Mill. "He possibly, therefore, went about it the wrong way, but in my view, his intentions were honourable enough, and a prosecution and a conviction in these circumstances is out of proportion to the seriousness of what he did." [20]

The Big Questions

SHOULD THE DISTINCTION EXIST?

The "no" argument—in other words, the argument for treating Web site security research/disclosure and software security research/disclosure in precisely the same way—is that there are no differences in the potential motives of the researchers, nor are there differences in the potential misuse of the research findings.

If a researcher discovers and/or discloses a critical, remote code execution vulnerability within the most recent version of Internet Explorer, it may be no more or less a risk to those affected than if a researcher discovers and/or discloses a critical vulnerability within a high-traffic Web site on which sensitive information is transmitted.

There is dispute over whether the potential *number of people* at risk is greater for a vulnerability within a popular Web site or one within an industry-leading, near-ubiquitous software application. Yet, regardless the quantity of those affected, the working group members agreed that a software vulnerability poses a generalized risk, diffused over a wide area, while a Web site vulnerability poses a highly localized risk, very grave to the site affected. Finding, disclosing and even writing an exploit for a vulnerability in a piece of software suggest that many computer systems can be attacked in such a way, but do not identify nor declare any particular server vulnerable. Disclosure of such a localized weakness as a Web site hole could be akin to providing attackers the blueprint for a direct exploit. Indeed there is question over whether merely finding the vulnerability is, in of itself, an exploit.

Again, the work done by a software researcher is conducted upon the researcher's own computer system. Any damage caused as a direct result of the experimentation (not necessarily a direct result of the discovered vulnerability) will be incurred upon the researcher's own computer system. Web site research, on the other hand, is conducted upon another party's Web server, and thus any unintentional denials of service or other damages are suffered by this other party.

This possibility of injury, plus the grave risk to individual site owners (and those whose private information is held by the sites) support the "yes" argument—that Web site security research/disclosure and software security research/disclosure should be legislated differently. This was the consensus opinion among working group members.

If a legal distinction must exist, should the difference be as wide as it now is, or should the law be changed to narrow the gap (presumably by softening regulations on Web research)?

Many members of the working group were in support of modifying the laws to some degree, because the work of Web researchers is unduly inhibited by the current legal climate. Several Web researchers within the working group said that even in the event that they found a hole quite by accident, they'd hesitate to disclose it even to the site owner, for fear of prosecution—this opinion grew stronger the more they learned during dialogue with working group members from the Department of Justice.

DOES THE LAW PROPERLY ASSIGN LIABILITY?

There are two main schools of thought on the question of who is liable for damage done via a vulnerability discovered and disclosed by an independent researcher. One says the burden is on the person who disclosed the vulnerability; one says the burden is on the owner of the vulnerable Web site.

One aspect of the former argument can be expressed—albeit oversimply—"if no one knows about it, it's not a vulnerability."

Thus the researcher should be liable for any damage done, directly or indirectly, by their research. Indeed, the U.S.'s Computer Fraud and Abuse Act seems to support this idea. The law's definition of "damage" is: "any impairment to the integrity of data, a program, a system or information [21]." The definition of "loss" is: "any reasonable cost to any victim, including the cost of responding to an offense, conducting a damage assessment and restoring the data, program, system or information to its condition prior to the offense [9] and any revenues lost, cost incurred, or other consequential damages incurred because of interruption of service [22]."

[9] It may be said, with some cheek, that the phrase "to its condition prior to offense" holds with it a bit of irony. Many of these possible "losses" will be incurred by the costs of repairing security vulnerabilities. If the system were truly being returned to its condition prior to the offense, it could be interpreted to mean that the system will be returned to its vulnerable state.

The latter argument counters that malicious parties will not be dissuaded by the law. They may have already found vulnerabilities without a tip-off from good-intentioned white hat hackers, and will not report these vulnerabilities to the site owner.

The latter argument further states that it is in the end user's best interest to be able to hold the Web site liable for breaches of their personal information, and that without due incentives to improve Web security, site owners are unlikely to invest more in securing their sites.

There are no legal penalties for being vulnerable, per se.

Should an organization experience a full-fledged data breach, however, there are costs—a mixture of shame and postage stamps—meted out by state laws; thirty-five states currently have data breach notification laws [23], many of which are based off of that of California.

The University of Southern California is subject to this law, which mandates that organizations that have experienced data breaches of a certain magnitude notify those individuals whose personal information may have been exposed. Thus, when the university learned of the flaw in their online application system, and learned that Eric McCarty had actually accessed and copied seven records, they were obliged to report the incident to all 275,000 applicants who may have been at risk.

When the criminal case was leveled against McCarty, the aim was to pin the liability on McCarty. Thus, the $140,000 in damages USC sought in the case, were mostly attributed to the postage costs of mailing the breach notification letters—the costs of repairing the site, and other costs related to bringing the site down for 10 days were also included in this figure, (despite the fact that the decision to take the application site offline for 10 days was an active choice made by USC; not a denial-of-service caused by McCarty). [14]

There was speculation within the working group as to whether this charge would have held, had McCarty only scanned the site and reported the SQL injection vulnerability to the university, rather than actually conducting a proof-of-concept exploit and accessing, copying and transmitting confidential records without authorization. Forensic investigation might have been able to prove that McCarty only accessed seven records, but no investigation into that matter was undertaken, and was moot once McCarty pled guilty.

Because McCarty signed a plea agreement, it is impossible to know how the court would have decided the case.

It is also unclear what liability would be assigned to a vulnerability discloser if an exploit of the vulnerability were later carried out by a criminal hacker; particularly if it could be proven, beyond a reasonable doubt, that the discloser was neither the exploiter nor an accomplice. No such cases have yet been tried.

DOES THE LAW UNDULY RESTRICT WEB RESEARCHERS' WORK?

Supporting the "no" argument is the idea that researchers do not need to dig into individual Web sites, unauthorized, in order to make significant strides in the field.

Indeed, without this freedom researchers have recently unearthed major weaknesses in Web security by studying common Web elements.

For example weaknesses were found in the JavaScript scripting language, allowing attacks like Javascript hijacking [24] and JavaScript malware used to hack into an organization's Intranet [25], as well as Jikto, a Web vulnerability scanner that uses JavaScript to employ an end user's browser to view a Web site through a proxy and thus, indirectly and undetected, can scan the site's data for vulnerabilities [26]. Google researchers analyzed content of several billion URLs, finding 450,000 that were successfully executing drive-by downloads of malware binaries [27]. Plus, the Internet Engineering Task Force (IETF) conducts ongoing research into the security of the domain name system (DNS) and other elements of the Web's underlying architecture.

Also, researchers are not altogether prevented from testing individual sites. If legally contracted by a site owner, they can perform penetration tests and maintain ongoing vulnerability monitoring on entire, live Web sites. From these experiences they can extrapolate insights about the general Net security concerns.

Indeed several Web security companies recently aggregated data from their customer bases and released figures detailing the prevalence of various vulnerabilities and the relative severity of these vulnerabilities. [28]

However the actual findings in research like this—for example that 85 percent of Web sites are vulnerable to cross-site scripting attacks [28]—could be interpreted to indicate that the general security of the Net is so poor that any measures excessively restricting the much-needed work of Web security researchers are not in the best interest of the Internet-using community.

Further supporting the "yes" argument is the assertion that the persistent disclosure of software security holes has strongly pushed the development community into creating more secure applications. If vulnerability disclosure could do this for software, could it not also do it for Web applications?

DO LAWS DAMAGE THE INTERNET?

Stemming from the argument that "yes, the laws unduly restrict Web researchers' work," is the question of how this situation affects the Internet itself.

Lost user confidence

Before handing sensitive information over via a Web application, Internet users can read Web sites' privacy policies, look for Web site security indicators (like VeriSign's Secured Seal and ScanAlert's HackerSafe) and request further information about an organization's security practices. Yet, they cannot personally test the security of the Web site.

The prevailing argument is that such a prohibition does not *directly* degrade most end users' confidence in Net security, because very few have the skill to actually conduct such tests.

Yet, there is evidence suggesting that the security-assessment options currently available to end users are inadequate.

A 2006 survey of Internet users found that only 29 percent of users said they could "strongly trust" the privacy policies of Web sites. Only half said they sometimes read the policies of sites. [29]

Other research shows that users do not understand what security indicators actually indicate [30]. (Indeed a wide variety of Web site security certifications exist, all of which provide assurance for something different, and most of which provide assurance for only a particular snapshot of time in the site's existence [31].)

Thus an argument follows that 1) user trust will only be bolstered when the Internet is worthy of such trust and when adequate assurance of that trust is available, and 2) that trust and assurance can be provided by the work of Web security researchers, and 3) thus, laws that prohibit Web researchers from testing Web sites' security *indirectly* degrade end users' confidence in Internet security.

And yet, while users' trust in Internet security is dropping [32] and the number of reported Web security incidents is increasing [33], Internet use for sensitive activities continues to increase [34].

So, whether or not Net security suffers, e-commerce remains a booming business.

Lost talent or lost liabilities?

Cuthbert has expressed opinions that the laws pose greater danger to security researchers than to criminals with malicious intent. As a result, Cuthbert chose to leave the security industry and his country. He now lives in Thailand and makes a career as a photographer. [6]

McCarty continues to work in the security field as an independent consultant. As he is still under house arrest, he has curtailed his pursuit of full-time employment, but he has previously encountered trouble getting hired because he has a felony on his record. [13]

Sixty-eight percent of respondents to the 2003 CSI/FBI Computer Crime and Security Survey said they would not employ a reformed hacker. [10] [35]

Some Web security researchers within the working group mentioned that they themselves got their start in the Web research industry by independently finding and disclosing vulnerabilities to site owners, not realizing at the time ("the time" being earlier than any of the case studies mentioned here) these actions might be grounds for litigation. Thus, they wonder if the current legal climate will prevent or inhibit young researchers from entering the field at all.

Damaged relations with law enforcement?

In 2005 Cuthbert posited that in convicting him, law enforcement shot itself in the foot. Many of his former colleagues in the security industry told him that the ruling will make them less willing to help law enforcement in their investigations of computer-related crimes, and less likely to report crimes. [6]

"Just by saying 'I found something suspicious,' they could get arrested just like me," he said. [6]

(Only 25 percent of respondents to the 2006 Computer Crime and Security Survey said they reported security incidents to law enforcement [36]. This was, however an increase from 2005, when only 20 percent reported incidents [37].)

SHOULD THE LAW CHANGE?

There are two principal elements to this discussion: whether the law should change what qualifies as "legal" or "illegal," and whether the law should establish different sentencing guidelines.

[10] CSI stopped asking that question with the 2004 survey, thus no more recent opinions are known.

The crimes information security professionals fret over daily may have techie terms like denial-of-service, phishing and port scanning—but ultimately the crimes themselves are the same ones law enforcement have been fighting for ages; crimes like fraud, theft, trespassing, extortion, blackmail, harassment, vandalism, destruction of property, etc. Really most of the terms we speak of as "cyber crime" are merely *methods* of crime that use computer systems.

Some argue that cyber crime law should be changed, so that it more accurately legislates the crimes, rather than the methods—first because these methods evolve so rapidly that laws will quickly become obsolete; second because this tight focus on methodology makes irrelevant the role of malicious intent.

Other arguments posit that adjustments to sentencing guidelines—either in addition to, or in lieu of changes to what is "legal" and what is "illegal"—will be more effective at addressing malicious intent.

CRIMINAL INTENT

Judge Ian Mill dismissed the case against Gerry Macridis on the basis that his "intentions were honourable enough, and a prosecution and a conviction in these circumstances is out of proportion to the seriousness of what he did." [20]

The words "out of proportion" are fundamental to this discussion, and the phrase inspires two different questions: Are the sentences that were meted out to these law-breakers in the cases above congruent to their crimes? Are the *potential* punishments for violating cybercrime laws congruent to the *potential* crimes?

Security professionals know better than anyone how severe the degree of damage can be from a cyber attack. Thus, members of this working force agreed that the laws should be powerful enough to soundly punish criminal hackers. Most members further assert that the laws must also be flexible enough to allow softer sentences for Good Samaritan Web security researchers.

Law experts on the working group pointed out that maintaining broad statutes provide judges more latitude when sentencing, and reduce obstacles when the case is considered particularly severe. Around the time of Cuthbert's conviction, revisions to the Computer Misuse Act were under discussion in the British Parliament—yet the suggested revisions allowed for more severe punishments for these crimes.

After the Cuthbert case, Derek Wyatt, a member of Parliament, and chair of the All-Party Internet Group (APIG) said, "It has never been the intention of myself or APIG that any revisions to the Computer Misuse Act would infringe upon security professionals' ability to do their job. However, with regards to Daniel Cuthbert, his 'job' was not to test the security of the Web site, but rather to determine whether the charity was in fact legitimate." Wyatt said Cuthbert could have searched for the DEC on the list of registered charities instead. [6]

Some members of the working group posited that cases like McCarty's and Cuthbert's show that law enforcement needs a "poster child" for computer crime laws, in order to validate the existence of such laws, or to prove their expertise in such cases.

In regards to Cuthbert's case, Detective Inspector Chris Simpson of the London Metropolitan Police Service's Computer Crime Unit stated "It is my firm belief that the conviction of Daniel Cuthbert will send out a powerful message to those who think hacking is some form of sport, where there is no victim, or indeed crime." [3]

Some members of the working group were not convinced that motive and malicious intent could be "proven," but attorneys within the group asserted that attorneys prove, beyond a reasonable doubt, criminal intent every day. Attorneys within the group said that "a good motive doesn't save you," and that the assessment often comes down to whether or not the jury thinks you're a good person.

One anecdotal example presented during working group discussion was of a defendant who wore to his hearing a T-shirt emblazoned with the logo for DefCon, a hacker convention.

Some within the working group expect that some of McCarty's actions would have brought his character into question as well. When he sent the first message to SecurityFocus, he did so under a pseudonym, from the e-mail address ihackedusc@gmail.com. In addition, McCarty reportedly posted a blog piece stating "USC got hacked. I was involved. I'm sorry, my bad. So all the hot USC girls, I got your phone number ladies. If your name is Amanda, Allison, Amy or Anita, expect a call any day now." [14]

DISCOVERY AND DISCLOSURE

So how can a court determine intent? The working group suggested that two main factors be assessed—the methods the individual used to discover the vulnerability, and the degree to which the insecure party was informed about the security weakness.

Discovery

The security professionals within this group developed a matrix of Web vulnerability research methods on a scale from least invasive to most invasive. The legal professionals then speculated upon how the law would most likely interpret these actions and the likelihood of prosecution. The latter half of this undertaking proved nigh impossible, being that this list could not properly take into account the long list of extenuating factors unique to each individual case that would influence these interpretations and actions.

The Web researchers within this working group collectively generated the following categorization of typical (and some less typical) Web vulnerability research methods. They fall into four basic categories, roughly ordered from least-invasive to most-invasive.:

❏ **Gather information off-site.** For example:
  - ✧ Scan public discussions for references to/ complaints about Web site that may indicate security weaknesses.
  - ✧ Search archive.org to learn about the site's history of technology adoption.
  - ✧ Conduct "Google hacking"—HTML sources and search results may reveal leaked data and tell-tale signs of vulnerabilities.
  - ✧ Social engineering—ask employees, ex-employees or tech support for details about site maintenance, known deficiencies, prior security incidents, etc.

❏ **Use site as normal, with an awareness of security.** For example:
  - ✧ Read site documentation, to learn of potentially insecure functionalities.
  - ✧ Read site's copyright and legal information for references to use of open-source code libraries.
  - ✧ Read any error messages that give away too much.
  - ✧ Go through the password reset process.
  - ✧ Check SSL certifications for weak ciphers or out-of-date certifications.
  - ✧ Time length of session life.
  - ✧ Look for patterns in use of session log identifiers.
  - ✧ Send HTML mail from site to own Webmail account, and then test message for cross-site scripting issues. [11]

❏ **Use site, purposely causing common errors.** For example:
  - ✧ Mistype username or password.
  - ✧ Submit empty forms or submit forms without all required fields.
  - ✧ Use bad formats or illegal values.

❏ **Use site performing functions with the intent of causing abnormal behavior.** For example:
  - ✧ Tamper with HTTP headers, looking for unblocked deep linking or image theft opportunities.
  - ✧ Conduct forced browsing—guess names for files or directories names that site owner did not intend to be publicly accessible. Search for source code, backup files, etc.
  - ✧ Conduct port scan.
  - ✧ Conduct ping sweep.
  - ✧ Sniff HTTP connections—investigate site's use of hidden fields and cookies; fingerprint the make, model, or version of the Web server.
  - ✧ Run vulnerability scan.
  - ✧ In any instance when access to sensitive data is gained, take screen shots.
  - ✧ In any instance when a vulnerability is found, perform a proof-of-concept exploit.

According to the representatives from the Department of Justice, no one of these actions alone would be a solid basis for pursuing prosecution. Building a stronger case would require a combination of these activities, plus other factors—–for example, if there's evidence that the defendant has attempted to cover his tracks, if damage was caused, what other collateral effects occurred, what actions the defendant subsequently took (such as disclosing the vulnerability, writing an exploit, asking for payment, selling the information, etc.).

Also adding complexity to the task of assessing prosecution likelihood, is the fact that there are many junctures at which significant decisions about the case are made—when the Web site owner decides whether or not to report the incident to law enforcement; when the State attorneys decide whether or not to prosecute; when the defendant decides whether or not to plead guilty; when the judge decides whether or not to dismiss the case; when the jury decides guilty or not guilty; when the judge passes sentence, etc.

In other words, it's very hard to say whether or not any particular action would lead to conviction.

Some security researchers within the working group were surprised to hear that some of these actions—particularly

[11] This "reflective XSS" method does not precisely fit into any of these broader categories, but was seen as important enough to include in this list.

running a vulnerability scanner like Nikto, taking screen shots, or writing exploits—(if done without authorization) were not, all on their own, sufficient grounds for prosecution. Further, some researchers suggested that law enforcement is unable to keep pace with the industry as far as technological advances go; thus they suspected that perhaps law enforcement inadequately understand the nuances that differentiate these research methods—for example, the difference in the invasiveness of a port scan and that of a vulnerability scan. A port scan is more of a "look-see," while a vulnerability scanner actually sends queries, with the purpose of invoking data.

The working group members raised the issue that if no one of these research methods in of themselves constitutes a solid case for prosecution, organized criminals may begin to separate research tasks to disguise their purposes and evade prosecution. However, the attorneys in the group said that if conspiracy is suspected, deep investigation and prosecution is highly likely.

These factors are not altogether clear in the physical world either, but perhaps clearer than they are in the logical world. Several physical-world scenarios, relatively analogous to logical-world scenarios, were discussed amongst the working group members. For example:

A man fears that his neighbor has left the door to her house unlocked while she's away. He's interested in the security of her house either because he's a Good Samaritan, or perhaps because he loaned her something valuable to him and he's afraid it hasn't safely been secured (see note on data ownership below). The man goes to her house and jiggles the door, finds it's locked properly and leaves.

Is this an attempt at unauthorized access? Is it different from the shell code command Daniel Cuthbert carried out?

What if this door-jiggler was actually a burglar in the process of "casing the joint," or what if a criminal hacker was conducting a port scan on one's network in preparation for a later attack? Do those info-gathering missions amount to a crime?

The CMA clearly assigns different degrees of punishment for information-gathering—"unauthorized access [4]" and "unauthorized access with intent to commit or facilitate commission of other effects [38]" are separate offenses. "Intent," however can be a slippery term to define.

How can the court tell the difference between a Good Samaritan (what the information security industry calls a White Hat hacker), a criminal (Black Hat hacker) and someone in between the two (the especially elusive Gray Hat hacker)?

One way to tell may be to assess the degree to which the insecure party was informed of the security weakness.

Disclosure

There are four basic levels of disclosure:

- ❑ **No disclosure:** Working group discussion of non-disclosure initially leapt to the idea that non-disclosure would be indicative of a black hat hacker gathering information in preparation for an attack. However, further discussion strengthened the sentiment that white hat hackers may disclose fewer and fewer vulnerabilities as they become more and more fearful of prosecution.
- ❑ **Disclosure to site owner alone:**
  - ✧ By a Good Samaritan white hat seeking no compensation;
  - ✧ By an individual seeking compensation (Macridis may appropriately be placed in this category). This particular situation is sticky—while the discloser may be a good-intentioned sort merely seeking reasonable compensation for their work, it could also be an ill-intentioned sort committing extortion.
- ❑ **Disclosure to "trusted" or "respected" third party:** In the case of McCarty, and other anecdotal cases discussed within the group, researchers disclosed vulnerability data to journalists, third-party business partners to the affected site owner, and other presumably respected sources.
- ❑ **Disclosure to public.** (See discussion of "Month of Search Engine Bugs" below, in subsection about extradition.)

In the opinions of the legal experts on the working group, a researcher enters "guessing land" and risks prosecution any time they disclose a vulnerability—they are largely at the mercy of the site owner. While some site owners may be appreciative, others may consider the research an unacceptable trespass and will report the matter to law enforcement.

Attorneys within the group said disclosure to the site alone could win the defendant some points, but it depends greatly upon other factors of the case.

Some site owners post disclosure policies, which can—depending on the nature of and the language of the policies—either make the legal situation clearer or more turbid.

Some policies request that researchers responsibly report vulnerabilities they find to the site owner alone. In fewer cases, site owners will even invite researchers to test the site. However, depending upon the language of the policy, many different outcomes are possible—though the policy may ask for the researchers to disclose flaws, they may not be able to prosecute if the researcher does *not* disclose it; if the researchers cannot understand the legal verbiage, they may violate the policy unintentionally; by having any policy at all, the site owner may weaken their case against a researcher.

DAMAGE

Under the U.K.'s Computer Misuse Act, the court can convict for unauthorized access or modification to computer systems, without proving that damage was done as a result of those actions. Daniel Cuthbert was convicted in this way. [4]

For certain charges under the U.S. Computer Fraud and Abuse Act, at least $5,000 worth of damage must be done for conviction. [10] Within the government there are some movements to have the law changed so that it does not stipulate the $5,000 floor. The rationale for such a change is that many damages of cyber crime are difficult to quantify in dollars and cents. Discussion above about McCarty's case raises a number of questions about the quantification of "damage," but whether or not one deems it appropriate to hold the vulnerability discoverer or discloser responsible for postage of breach notifications, there is at least a bill from the post office stating exactly how much that cost was.

Yet what is the price tag on the privacy of an individual whose personally identifiable information was exposed through an independent Web site researcher achieving unauthorized access to a Web-based database? [12]

EXTRADITION TREATIES

In each of the case studies described above, the defendant was tried in his own country of citizenship, under the laws of that country, for crimes committed while in that country, against organizations that also operated in that country.

Yet such clear decisions on jurisdiction are rare reprieves in Internet law. The already muddy waters of international jurisdiction are further muddied by the hard-to-define boundaries of the Internet.[13] Thus, in addition to alterations to national law and sentencing, it is suggested to examine extradition treaties between nations.

"UFO hacker" Gary McKinnon has still not been tried for the charges brought against him in 2002, because McKinnon, a citizen of the U.K. has been fighting tooth and nail, via a string of extradition hearings and appeals, being tried in the U.S. McKinnon is charged with multiple counts under the Computer Fraud and Abuse Act for hacking into sites at NASA. The latest action was that McKinnon lost his appeal, and will now be tried in the United States. [39]

The United States signed a new extradition treaty with the United Kingdom in September of 2006 [40]. How this new treaty will affect McKinnon's case remains to be seen.

Macridis' case clearly shows that New Zealand law vastly differs from that of the U.S. and the U.K. The United States' extradition treaty with New Zealand has not been updated since 1970 [41].

A Ukrainian vulnerability researcher, who goes by the name MustLive, has declared June 2007 the "Month of Search Engine Bugs," ensuing as this report is being written. In response to questions about the legality of his research, MustLive responded (albeit in slightly broken English) that he was a citizen of Ukraine, was closely familiar with the laws of Ukraine, was not in violation of any laws in Ukraine, and thus was not in any danger. [42] None of the leading search engine companies have yet publicly threatened litigation.

It is unclear whether or not MustLive could break laws of other countries and face extradition. The Council of Europe's treaty on cybercrime, which has been signed and ratified by the Ukraine [43], lays out rules about jurisdiction [44] and extradition [45] that aim to resolve some of these questions. The U.S. has also signed and ratified the treaty as a non-EU-member [43]; the U.S.'s individual extradition treaty with the Ukraine was last updated in 2000 [46].

Symantec's most recent Internet Security Threat Report (March 2007) presented several top-10 lists, ranking countries by the amount of malicious activity originating in them [47]. Three of these countries—China, Taiwan and Bulgaria—have no extradition treaty with the United States [48].

[12] Do users even have a right to protect their own personal information? Laws on data ownership do not appear to consider individuals the owners of their personal data. Rather the steward of the data is the owner (for example a data broker like Choicepoint).

[13] Perhaps a useful analogy is to international environmental policy. The Internet, like the ozone layer, is a global, public resource, and actions done on one nation's soil (i.e. chlorofluorocarbon emissions) can do detriment to other nations (i.e. ozone holes over Antarctica).

What Can the Industry Do in the Meantime?

The initial question of this report is whether the law unduly restricts Web researchers' work. But at the end of the day, the point is to make a more secure Internet.

Technological advances and recent discoveries have added immediacy to this challenge. More and more Web sites have attired themselves in Web 2.0 apparel—online auctions, social networking, live streaming video, and the like. Sites that invite user-contributed content already tend to suffer from security issues. Plus, the nimble, responsive, really-listens-to-you nature of many Web 2.0 sites relies upon AJAX (Asynchronous JavaScript And XML). Unfortunately, a number of recent findings have shown that AJAX suffers from critical vulnerabilities, and that JavaScript can be used for a host of malicious purposes. [26]

What this means is that Web 2.0 hasn't just inherited the same frailties of Web 1.0, but suffers from new vulnerabilities we've not yet had to battle. Further, by giving more responsibility and more power to AJAX apps, Web 2.0 makes possible more sophisticated and severe exploits, using the same old Web 1.0 vulnerabilities.

Users' mitigation options are few and undesirable. Securing Web 2.0 is partly the duty of Web browser developers, partly the duty of security pros, and mostly a job for Web application developers.

However, due to legal strictures, compared to software developers, Web site developers do not as significantly benefit from, (nor chafe at), the security community's detection and disclosure of vulnerabilities within their applications.

So, all things being equal, no changes to the legal climate whatsoever, what can the security industry now to improve Web security without defying legal constructs?

Industry Regulations and Standards

Several within the working group, and the security community at large, assert that the need to comply with various regulations—Sarbanes–Oxley, HIPAA, Basel II, Gramm-Leach-Bliley, data breach notification acts etc.—has impelled organizations, if not necessarily to improve security, to invest more in and become more aware of security. [36]

In addition to national and state legislation, some organizations must also comply with industry regulations.

Arguably, foremost of these is the Payment Card Industry's (PCI) Data Security Standard (DSS), which applies to any financial organization that issues credit cards, any financial organization that acquires contracts with merchants that agree to accept credit cards as payment and any business that accepts credit cards as payment. [49]

The PCI-DSS is arguably the most rigorous collection of security mandates within both the industry and the government. It contains regulations about encryption of data transmitted over networks, two-factor authentication for remote access, and other network security. Yet the PCI-DSS standards do not directly deal with the secure Web application development raised in recent research. [49]

The working group discussed the need for better standards for secure Web site development.

August 31, 2006 the National Institute of Standards and Technology (NIST) released the public comment draft of special publication 800-95, "Guide to Secure Web Services." Since that time more widely publicized vulnerabilities within JavaScript and exploits written in JavaScript have been found. In the 140-page document, the word JavaScript never appears. [50]

Better Channel for Disclosure

Many Web security experts within the group fear disclosing vulnerabilities they've found, lest they face law suits. One suggested way of assuaging researchers' fears and encouraging disclosure is to create an anonymous disclosure line—similar to the anonymous tip lines law enforcement and intelligence agencies use to gather critical criminal information they might otherwise not receive from known informants.

Dummy Pages

Another option is for Web site owners to construct a "dummy" of their Web site—a site that mirrors the architecture of their Web server, but contains only "dummy" (phony) data. Inviting researchers to crack into that site might provide useful insight, without putting any real data at risk.

There are some problems with this idea. If this dummy site existed on the same server as the legitimate one, it might be possible to hack into the dummy site and then crack into the legitimate site from the inside. One modified suggestion proposed in the working group was to set up the dummy site on a separate, quarantined server.

Yet, if the dummy site too closely resembled the real site, it might be trivial to nab the real data by simply repeating one's steps on the real site.

Thus a modified idea is for a third-party entity to create a series of dummy sites that mimic typical sites in a variety of categories—social networking, banking [14], e-commerce, government, small business, etc. Web vulnerability researchers would then be invited to hack the sites and report their findings. There was some discussion but no consensus within the working group regarding compensation for disclosers. Consensus did exist on the opinion that all contributors must be registered with the site, but not on whether registrants' access would be extended/restricted on the basis of certain prerequisites, such as experience in the field, professional certifications, etc [15].

Though a project of this kind would not detect problems within an individual site, it might be fodder for better standards for Web developers, better Web site security certification services, etc.

Yet, if such a third-party service existed, would anyone use it? The less a dummy site resembles the real site, the less valuable the findings are. If the site owners had to pay to learn the results of such a service, the cost would have to be significantly less expensive than the costs of hiring a professional penetration testing or vulnerability monitoring service.

Conclusion

The meeting of minds that took place over the past two months advanced the group's collective *knowledge* on the issue of Web security research law. Yet if one assumed that the discussion advanced the group's collective *understanding* of this issue, one might be mistaken.

Informative though the work was, it raised more questions than answers. In the pursuit of clarity, we found, instead, turbidity.

Thus it follows, that there are many opportunities for further thought, further discussion, further research and further stirring up of murky depths. In the short term, the working group has plans to pursue the following endeavors:

❏ Creating disclosure policy guidelines—both to help site owners write disclosure policies, and for security researchers to understand them.

❏ Creating guidelines for creating a "dummy" site.

❏ Creating a more complete matrix of Web vulnerability research methods, written with the purpose of helping attorneys, lawmakers and law enforcement officers understand the varying degrees of invasiveness.

These goals may be added to or changed. The next report will be released Nov. 5, concurrently with CSI's Annual Conference and Exhibition. Periodic updates will appear at GoCSI.com.

[14] Foundstone, Inc. created a program called Hacme Bank, designed to teach application developers, programmers, architects and security professionals how to create secure software. Hacme Bank simulates a "real-world" Web services-enabled online banking application, which was built with a number of known and common vulnerabilities. The intended benefit of this application is to offer users a chance to attempt real exploits against a Web application and thus learn the specifics of the issue and how best to fix it.

[15] Perhaps the EC Council's Certified Ethical Hacker certification.

References

1. Sara Peters, "Patcher Beware: What's riskier? The zero-day exploit or the unofficial patch?" CSI *Alert,* December 2006.
2. Zeroday Emergency Response Team (ZERT), http://zert.isotf.org/
3. Oct. 6, 2005, Press release, Metropolitan Police Force, Computer Crime Unit
4. Computer Misuse Act 1990 c. 18, Sec. 1, http://www.opsi.gov.uk/acts/acts1990/Ukpga_19900018_en_2.htm
5. Computer Misuse Act 1990 c. 18, Sec. 3, http://www.opsi.gov.uk/acts/acts1990/Ukpga_19900018_en_2.htm
6. Sara Peters, "Does U.K. Conviction Threaten InfoSec Pros?" CSI *Alert,* November 2005. Some information in article comes from exclusive interviews with Daniel Cuthbert.
7. Computer Misuse Act 1990 c. 18, Sec. 1, (1)(c) http://www.opsi.gov.uk/acts/acts1990/Ukpga_19900018_en_2.htm
8. 18 USC, Part I–Crimes, Chapter 47–Fraud and False Statements, Sec. 1030–Fraud and related activity in connection with computers, (e)(2)(B). http://uscode.house.gov
9. 18 USC, Part I–Crimes, Chapter 47–Fraud and False Statements, Sec. 1030–Fraud and related activity in connection with computers, (a)(5)(A)(ii). http://uscode.house.gov
10. 18 USC, Part I–Crimes, Chapter 47–Fraud and False Statements, Sec. 1030–Fraud and related activity in connection with computers, (a)(5)(A)(i)(B)(ii). http://uscode.house.gov
11. Plea Agreement for Defendant Eric McCarty, Sep. 5, 2006, United States District Court for the Central District of California
12. Robert Lemos, "Breach case could curtail Web flaw finders," *SecurityFocus,* April 26, 2006. http://www.securityfocus.com/news/11389
13. Sara Peters, "Does the Law Restrict Web Researchers' Work?" CSI *Alert*, May 2007. Some information comes from exclusive interviews with Eric McCarty.
14. April 2006, Affidavit filed by Special Agent Geoff Bickers, Federal Bureau of Investigation.
15. Robert Lemos, "Flawed USC admissions site allowed access to applicant data," *SecurityFocus*, July 6, 2005. http://www.securityfocus.com/news/11239
16. California Civil Code 1798.29–Notice of Security Breach Law, amended by California Senate Bill 1386 California Information Practice Act, 2002, which added Section 1798.82 to Civil Code.
17. California Civil Code, Section 1798.82 (e)(3).
18. Summary of facts, Damian Murphy, Detective Constable, Wellington Central CIB. Portions of this document were redacted in version sent to author of this report.
19. New Zealand Crimes Act 1961, Sec. 252
20. Oral judgment of Judge I. G. Mill, *New Zealand Police vs. Gerasmios Macridis,* Sep. 27, 2006, District Court at Wellington.
21. 18 USC, Part I–Crimes, Chapter 47–Fraud and False Statements, Sec. 1030–Fraud and related activity in connection with computers, (e)(8). http://uscode.house.gov
22. 18 USC, Part I–Crimes, Chapter 47–Fraud and False Statements, Sec. 1030–Fraud and related activity in connection with computers, (e)(11). http://uscode.house.gov
23. See the National Conference of State Legislatures, http://ncsl.org/programs/lis/cip/priv/breachlaws.htm.
24. Brian Chess, Yekaterina Tsipenyuk O'Neil, Jacob West, "JavaScript Hijacking," March 12, 2007. www.fortifysoftware.com/servlet/downloads/public/JavaScript_Hijacking.pdf
25. Jeremiah Grossman, "Hacking with JavaScript Malware," CSI NetSec '07 conference, Scottsdale, Ariz. June 11, 2007.
26. Sara Peters, "AJAX and Hijacks: Web 2.0 is growing up. And we're not ready." CSI *Alert,* May 2007.
27. Niels Provos, Dean McNamee, Panayiotis Mavrommatis, Ke Wang and Nagendra Modadugu—Google, Inc., "The Ghost in the Browser: Analysis of Web-based Malware," May 2007. www.usenix.org/events/hotbots07/tech/full_papers/provos/provos.pdf
28. Web Application Security Consortium, Statistics. http://www.webappsec.org/projects/statistics/
29. Gideon, Cranor, Egleman and Acquisti, 2006.
30. Nan Poulios, "The Failure of Security Signals," CSI *Alert,* January 2007. Modified from original printing in Peltier and Associates "Year in Review, 2006."
31. Rebecca Herold, "Do 'Hacker-safe' Certs Give SMBs a False Sense of Security?" CSI *Alert,* April 2007.
32. Cyber Security Industry Alliance, Digital Confidence Index, Spring 2006. https://www.csialliance.org/publications/publications/surveys_and_polls/dci_survey_May2006/

33. Privacy Rights Clearinghouse, "A Chronology of Data Breaches." http://www.privacyrights.org/ar/ChronDataBreaches.htm
34. Antony Savvas, "Online Banking with HSBC Grows," *ComputerWeekly.com,* May 30, 2007. http://www.computerweekly.com/Articles/2007/05/30/224388/online-banking-with-hsbc-grows.htm
35. Robert Richardson, *2003 CSI/FBI Computer Crime and Security Survey.*
36. Lawrence A. Gordon, Martin P. Loeb, William Lucyshyn and Robert Richardson, *2006 CSI/FBI Computer Crime and Security Survey.*
37. Lawrence A. Gordon, Martin P. Loeb, William Lucyshyn and Robert Richardson, *2005 CSI/FBI Computer Crime and Security Survey.*
38. Computer Misuse Act 1990 c. 18, Sec. 2, http://www.opsi.gov.uk/acts/acts1990/Ukpga_19900018_en_2.htm
39. "UK hacker loses extradition fight," *BBC News*. April 3, 2007. http://news.bbc.co.uk/2/hi/uk_news/6521255.stm
40. Treaty Number 108-23—Extradition treaty with the United Kingdom. Sep. 29, 2006. http://thomas.loc.gov/home/treaties/treaties.html
41. Treaty Number 91-13—Extradition Treaty with New Zealand, May 27, 1970. http://thomas.loc.gov/home/treaties/treaties.html
42. MustLive, "Month of Search Engine Bugs." Websecurity. May 15, 2007. http://websecurity.com.ua/955/
43. Council of Europe Convention on Cybercrime, CETS No. 185 http://conventions.coe.int/Treaty/Commun/ChercheSig.asp?NT=185&CM=2&DF=6/8/2007&CL=ENG
44. Council of Europe Convention on Cybercrime, CETS No. 185, Chapter II, Section 3, Article 22, http://conventions.coe.int/Treaty/en/Treaties/Html/185.htm
45. Council of Europe Convention on Cybercrime, CETS No. 185, Chapter III, Section 1, Article 24, http://conventions.coe.int/Treaty/en/Treaties/Html/185.htm
46. Treaty Number 106-16—Treaty with Ukraine on Mutual Legal Assistance in Criminal Matters, Oct. 18, 2000. http://thomas.loc.gov/home/treaties/treaties.html
47. *Symantec Internet Security Threat Report: Trends for July–December 06*, Volume XI, March 2007. http://www.symantec.com/enterprise/theme.jsp?themeid=threatreport
48. Library of Congress Treaty database. http://thomas.loc.gov/home/treaties/treaties.html
49. Payment Card Industry Data Security Standard. https://www.pcisecuritystandards.org/tech/index.htm
50. Anoop Singhal and Theodore Winograd, "Guide to Secure Web Services (DRAFT): Recommendations of the National Institute of Standards and Technology, Aug. 31, 2006. http://csrc.nist.gov/publications/drafts.html