Ethical Hacking: The Security Justification Redux

Bryan Smith William Yurcik David Doss
Illinois State University
{basmit3 wjyurci,dldoss}@ilstu.edu

Abstract

The state of security on the Internet is bad and becoming worse. One reaction to this state of affairs is a behavior termed "Ethical Hacking" which attempts to proactively increase security protection by identifying and patching known security vulnerabilities on systems owned by other parties. Ethical hackers may beta test unreleased software, stress test released software, and scan networks of computers for vulnerabilities. Previous work has emphasized ethical hacking as an altruistic behavior but we find ethical hackers act rationally, in self-interest, to secure systems that are within their own community (sometimes for pay) – networked systems are only as secure as the weakest system within perimeter defenses.

1. Introduction

Social implications accompany technological advances. Social change resulting from technological advances may manifest itself in the changing perceptions of self, shifting definitions of moral behavior, and increasing demands for protection from newly perceived dangers.

This paper examines a social behavior rooted in the poor state of information security on the Internet that was first documented in 1987 [1]. Ethical hackers believe one can best protect systems by probing them while causing no damage and subsequently fixing the vulnerabilities found. Ethical hackers simulate how an attacker with no inside knowledge of a system might try to penetrate and believe their activities benefit society by exposing system weaknesses – stressing that if they can break these systems so could terrorists. The result is not only enhanced local security for the ethical hacker but also enhanced overall Internet security.

Hacking is a loaded term - the distinction between hacking and cracking is not universal. The concept of hacking is derived from the dictionary meaning of "hack" as a verb "to chop or cut roughly, to make rough cuts" as in programming using ad hoc methods based on experience without necessarily having a formal plan or methodology for evaluation [4]. While hacking has in the past been considered as counter-cultural, this is changing. Hacking may have been counter-cultural at one time but it was never anti-social since the result of hacking is a "*hack*" (a clever but unstructured programming solution to a problem) that can only be realized if it is shared with others – there is no such thing as a "private hack" [4].

Unauthorized computer intrusions are considered illegal in all but the most desperate of circumstances.[1] Once hacking ability is used to commit a crime the hacker becomes a criminal [9]. Criminal hackers or "crackers" gain unauthorized access primarily to seek financial gain but recently other motivations of crackers have been categorized such as seeking to subvert systems, doing damage to systems (vandalism), promoting political causes (hactivism), and acting as an agent of a foreign state (cyberterrorism and information warfare). The misapplication of the term cracker to a law-abiding hacker is due to celebrated incidents of unauthorized intrusions into computer systems that have incorrectly been attributed to hackers due to the extensive programming skill needed to achieve success. In this paper we will maintain this distinction, the term hacker to mean a law-abiding programmer of special characteristics and cracker to mean a criminal programmer.

When "ethical" is placed in front of the term hacking it denotes moral activity. Unethical hacking has no permission to intrude on systems. Ethical hacking includes permission to intrude such as contracted consulting services, hacking contests, and beta testing. If there is no permission to intrude, ethical hackers still find ad hoc ways to become aware of the system security of other systems. The end goal of ethical hackers is to learn system vulnerabilities so that they can be repaired for community self-interest - and as a side-product also the common good. Networked systems are dependent upon each other for system security so awareness of the security of machines within one's community-of-interest is not entirely altruistic but rather concerned with system security.

This paper is an extension of [9] emphasizing the self-interest motivation of ethical hackers over, but not excluding, altruism. The rest of this paper is organized as

0-7803-7824-0/02/$10.00 ©2002 IEEE.

[1] Spafford states that computer intrusions are ethical only in life-saving circumstances [10].

follows: Section 2 illuminates the problems of Internet security. Section 3 focuses on the primary technological tool of both ethical hackers and crackers - scanners. Section 4 describes the ethical hacking behavior of penetration testing. Section 5 addresses the specific issue inherent to all forms of security products and services – conflict-of-interest. Section 6 is a summary discussion of ethical hacking and we end with conclusions in Section 7.

2. The Problem

The Internet has become indispensable to business by allowing organizations to conduct Ecommerce, provide better customer service, collaborate with partners, reduce communications costs, improve internal communications, and access needed information rapidly. While computer networks have revolutionized the way businesses operate, the risks they introduce via interconnectivity can be devastating. Attacks on computer systems via the Internet can lead to lost money, time, products, reputation, sensitive information, and lives. In the rush to benefit from using the Internet, organizations have often not come to terms with significant risks including:

- Time-to-market pressures are forcing vendors release products too early with inadequate or no testing. The impact of defective software is immense; causing firms to lose nearly $100 billion last year in repair costs, downtime, and lost productivity [8].
- Current software engineering practices used by vendors do not produce systems that are immune from attack.
- System operators do not have the people or best practices to defend against attacks or minimize damage.
- Policy and law in cyberspace is immature and lags the state-of-the-art in attacks.
- There is a continued movement to complex, client-server, and heterogeneous configurations with distributed management.
- There is little evidence of security improvements in most systems since new vulnerabilities are routinely discovered.
- Current security tools are lacking in that they only address parts of the problem and not the system as a whole. Lack of understanding leads to reliance upon partial solutions.
- System administration is difficult and becoming unmanageable due to patching against increased vulnerabilities.

As if the situation needed to be any worse, intruders are building a growing technical base of knowledge and skills leveraged through automation and exploiting network interconnectivity.

In response, the market for security products and services is growing faster than the supply of quality products and service providers. Consumers need to go beyond awareness to critical understanding but urgency has also created many problems – products and services have moved to this niche unfortunately only selling snake oil - "If you want it badly, you'll get it badly". It is becoming a consensus that there is no single product or group of products that can be bought to create security but rather a combination of products with skilled personnel and business processes.

The end result is a "wild west" scenario where the average time for a PC to be broken into directly out-of-the-box from the store and attached to the Internet is less than 24 hours with a worst case scenario of 15 minutes.[2] Many, who can afford it, are turning to "hired guns" for protection. To continue this western metaphor, the "town sheriff" who maintains community protection is the ethical hacker.

3. Scanning Tools

A port is an interface where different layers of software exchange information. Port scanning floods messages to a large number of ports to gather responses to be analyzed. Typical information that can be learned from a port scan is the existence of a computer, the operating system, version of operating system, types of services available (smtp, httpd, ftp, telnet, MSWindows networking services, etc.) and type of computing platform. A representative list of some scanners is shown in Table 1.

In technical detail, a scanner sends a message requesting to open a connection with a computer on a particular port. The computer has the option of ignoring the message, responding negatively to the message, or opening a session. Ignoring the message is the safest since if there are no open services it may be hard for a cracker to determine if a computer exists. Responding to all message requests, even negatively, provides information that can be used maliciously (knowing a computer does exist – concentrating on finding a way in). Once a port scan reveals the existence of an open service, a cracker can attack known vulnerabilities.

All services at some point in their lifetime will have known vulnerabilities. There is no way to connect usefully to the Internet and not have some risk from this exposure. In fact hundreds of new vulnerabilities are found annually and all it takes is one un-patched vulnerability for a system to be completely compromised.

[2] Data from Bruce Schneier "Schneier on Security" Chicago IL March 20 2002 which is also supported by data from The Honeynet Project <http://project.honeynet.org/>.

Once a cracker scans all computers on a network and creates a network map showing what computers are running what operating systems and what services are available, almost any kind of attack is possible including automated scripted program attacks and social engineered attacks given the detailed knowledge obtained.

Initially written for Unix platforms, the first port scanners were freeware scripts designed to help crackers in their reconnaissance of potential victims by identifying open ports and vulnerable services. With the advent of Windows port scanners, primarily intended to test security policy compliance, this fixture in a cracker's arsenal is gaining acceptance as useful defensive tool.

The first scanner to gain notoriety was the Security Administrator's Tool for Analyzing Networks – SATAN introduced by Dan Farmer in 1995. Unlike previous automated scanners that ran on the particular system being analyzed, SATAN could analyze any system accessible over the Internet [9]. Proponents of SATAN viewed it as a system administration tool to find security flaws in order to prevent intrusions. Opponents of SATAN viewed it as an easy-to-use tool an inexperienced programmer (script kiddie) could use to bring down systems all over the Internet [9].

To explore this dual nature of a scanning tool I will use the following analogy, port scanning is equivalent to looking at doors to gain information about what the site does [2].[3] This is useful since the functionality and vulnerabilities of an Internet presence is determined by its operating system and computing platform. Crackers commonly refer to determining the details of an Internet presence as a "pre-attack" [14].

Table 1. Representative Scanning Tools

commercial scanners	ISS- *SafeSuite* Network Assoc-*Cybercop*,	
freeware scanners	SARA/NESSUS; AINT	
sniffers	SNORT(free), TCPDUMP(free)	records packets
network scanners	Nmap;NSS;SATAN;ISS Strobe;Trinux;NetSonar	TCP/UDP/PING sweeps
war dialers	THCScan (free)	
password crackers	JohntheRipper; Netective	
firewall scanners	Firewalk	determines ruleset
host port testers	NetCat, HackerWhacker ShieldsUp	connects to open ports & proxies
host OS scanners	Unix netstat – shows active sockets for each protocol & displays packet traffic on network interfaces; Cops; Tripwire – checks system changes; check.pl	
scan detectors	PortSentry,Hping	

Why should anyone with an Internet presence and no interest in cracking other systems learn about scanners? The answer is to learn what crackers will see in their own Internet presence since scanners are common attack starting points. Crackers look for unauthorized[4] services such as someone running a server with known problems, an unauthorized server on a high port, or unauthorized server on a legitimate port [2]. Port scanning can be done manually from a single computer to learn about target systems or it can be done automatically by program(s) originating from multiple computers on different networks to a large number of target systems (breadth) or from it can be done automatically by program(s) originating from multiple computers on different networks to a single target system over a long period of time (depth).

Preventing security flaws from being discovered via scanning is twofold: (1) eliminate potential problems by attempting to keep systems up-to-date with security patches and if it is critical to run dangerous services, limit exposure via a firewall and (2) monitor traffic to detect and block scans. Scans have been detectable by the serial nature of messages to sequential ports but scanners have evolved to scan ports in random order.

Let's reconsider our door metaphor for port scanning. If you check the doors of your own house to see which are locked there is no problem but what if you check the doors of all the houses in your neighborhood to see which are unlocked? Is it illegal if you do not open the door? The US Federal District Court of Georgia has decided that the act of port scanning did not constitute "damage" to a network that would trigger liability under both Georgia's computer fraud laws and the Federal Computer Fraud and Abuse Act. The two major adversary arguments were: (1) port scanning is a common and necessary function of Internet operations – all web browsers scan port 80 on every URL typed into the location bar versus (2) unauthorized scans should be illegal just as "casing" a neighborhood or parking lot is illegal.

Port scanners, like other tools, have both offensive and defensive applications – what makes a port scanner good or evil is how it used. We mention this here because a port scanner is simultaneously both the most powerful tool an ethical hacker can use in protecting a network of computers and the most powerful tool a cracker can use to generate attacks. Historically the most popular cracker attacks are those that use scanning tools to target known vulnerabilities.

[3] Bishop's door examples include: half-door with counter=fast food; heavy glass with sliding tray=bank teller; door with iron bars=jail.

[4] Authorized services are usually protected by an organization's perimeter security defenses while unauthorized services can be unknown to security manager who must be aware to protect it.

4. Penetration Testing

The idea of testing the security of a system by trying to break into it is not new. This type of testing is notably used to determine automobile crashworthiness as one example. The earliest work on penetration testing in computer systems dates back to 1975 [5].

Penetration testing is not sufficient by itself – passing a penetration test does not mean the tested system cannot be compromised [6]. The penetration tests are often only as thorough as the people administering them so known vulnerabilities may be missed. Scans have been known to miss important "pop-up" servers that periodically connect and then quickly disconnect from the network. Since scanners only check for known vulnerabilities, a system that successfully passes a scan may still be wide open to a new unknown attack.

Penetration testing by ethical hackers is among the most thorough methods for finding vulnerabilities and increasing protection for a dynamic network of computers. Correctly performed, a penetration test is a covert test in which a paid consultant or ethical hacker plays the role of a hostile attacker who tries to compromise system security. Since the ultimate goal is penetration, the test is carried out without warning – ideally upper management has approved the test.

Incorrectly performed, penetration testing also has a potential for creating damage. While other types of testing are usually performed cooperatively with an organization's staff, damage caused by penetration testing may go unnoticed for some time. Active scanning can be very disruptive since some computers are fragile and do not handle port scanning well. Database servers and mainframes are notorious for being crippled by tools such as ISS Scanner and NMAP.

Crackers routinely scan networks of computers for security flaws that can be exploited (exploits) and then post this sensitive information on the Internet for others to take advantage of. This is one reason why ethical hackers regularly browse known cracker websites and mailing lists to monitor cracker activity. Finding security flaws before crackers do lowers the risk exposure of an organization:

- Even a single incident could cost significantly – both financial and reputation damage.
- It reduces vulnerabilities and points of intrusion.
- A tight system reduces the probability of attack – the attackers will go to easier and more attractive targets.
- An on-going program lowers insurance rates.

Penetration testing using ethical hacking provides both assurance and insurance: assurance that the given environment will resist attack and insurance that the organization is acting in a prudent manner. Because penetration testing invariably ends up discovering security holes on client networks/computers, most clients do not want to talk on record about the results of such tests. However, numerous generic examples exist where penetration testing has saved businesses embarrassment and loss of reputation:

- Online services organization always tested prior to new releases.
- Financial institutions saved embarrassment prior to release of a new online brokerage offering.
- Another financial institution has a policy of testing before any Internet application goes live.

Penetration testing is an accepted technique. The National Institute for Standards and Technology (NIST) has recently released a document describing a methodology for using network-based tools for testing systems for vulnerabilities [11]. The primary aim of the document is to help system administrators get started with a program for testing on a routine basis. The methodology recommends focusing first on those systems that are accessible externally (e.g. firewalls, web servers) and then moving on to other systems as resources permit.

Computer-security services, including penetration testing by ethical hackers and other services, was a $1.8 billion world-wide market in 2001 and is expected to grow at a compound annual rate of 28% for the next three years [12]. Organizations ranging in size from startups (HackerWhacker, SPI Dynamics-Webinspect, Sanctum-AppScan) to subscription scanning services (Network Associates, World Wide Digital Security Inc.) to corporations (Predictive Systems, Computer Science Corporation, and IBM), to the US Department of Defense have ethical hacking teams (sometimes also called tiger teams or red teams)[12,6,3,13].

Although ethical hacking is an effective measurement tool and a crucial component of any security program, it should only be part of a larger security program. A comprehensive security program incorporating ethical hacking can be used to discover and correct frequent errors early in the design, implementation, and test process which shortens development time and cost. Ethical hackers provide feedback to system designers and discover problems that may otherwise go undetected. The problem is that crackers can do their own penetration testing and do it more frequently. The best a penetration test can do is to provide a snapshot in time. Periodic testing is necessary to ensure compliance against a baseline. Tools are evolving to do continuous monitoring of security configurations.

5. Conflicts of Interest

"Ethical Hacking" has been widely marketed as an essential tool in information security but there are obvious conflicts of interest.

Security firms have an incentive to hype threats and invent threats. As the market potential has grown, unscrupulous vendors have been quoted overstating dangers to expand customer base and in some cases selling products that may actually introduce more vulnerabilities than they protect against.

Convicted criminals can earn large salaries working on "ethical hacking teams" while simultaneously supporting software tools designed to exploit vulnerabilities in commercial products ostensibly to "illustrate the seriousness of the problem" or to "promote vendors taking security seriously [9]. Some individuals who work at security firms have been known to spend their *off-hours creating and distributing the very attack tools* their company sells products to protect against. It is important to realize that sensitive data will be exposed during penetration testing creating dangerous insider threats.

Lastly, in actions accentuated by market pressures, businesses have used ethical hackers to:

- beta test new products - stress testing and reporting back information about defects in pre-release software in exchange for early access to this new software
- hacking contests – Argus, Lucent, and Oracle (to name a recent few) have held "cracking" publicity contests offering prizes for an intrusion into one of their products

There are large problems with the effectiveness and efficiency of both of these activities but setting that aside for the moment, the basic premise is the use of ethical hackers to harden software that has not been adequately tested. There is conflict-of-interest in that businesses do not want to redevelop software that should have incorporated security testing throughout its entire development so these activities are superficial at best. There is also hypocrisy in that businesses are encouraging cracking behavior that they would prosecute under any other circumstances.

6. Discussion

In [10] the argument is made that the security justification for ethical hacking is flawed in two ways: (1) exposing security flaws should not be encouraged or rewarded and (2) not every organization has the resources to maintain current versions and patches on their system software. While it may not been as clear in the past, networked systems (especially in communities-of-interest) are clearly now dependent upon each other for security. Just one insecure machine within a large network can be used as a platform upon which to launch attacks. The distributed denial-of-service attacks of February 2000 using compromised machines to indirectly flood E-commerce sites are a recent example of this interdependence. Thus each computer's security is dependent on the security of other computers within its community-of-interest such that exposing security flaws is a positive action in both self-interest and common good.

With the present poor security on the Internet, ethical hacking may be the most effective way to proactively plug security holes and prevent intrusions. On the other hand, ethical hacking tools (such as scanners) have also been notorious tools for crackers. A fine line exists between hacking for the community interest and public good versus releasing tools that may actually enable attacks and in aggregate make the Internet less secure when taken as a whole [9].

7. Summary

Hacking has entered the age of mass production. Current and future Internet attacks are a technologically enabled crime - shifting from manual to automated attacks. Automated scanning tools as a pre-attack tool are a substantial threat to the Internet – a few widely available automated tools endanger the majority of Internet-based computers. Ultimately the solution to automated attacks is more effective defenses based on new technology in some cases and the law for prosecution in some cases. We cannot eliminate cracking through solely technical or legal means but until the future solution what are we to do in the meantime?

Security used to be a private matter. Until recently information security had been left largely in the hands of a few specially trained professionals. The paradigm shift of technologically enabled crime has now made security everyone's business. Ethical hackers see this clearly and are responding to actual threats to themselves and in the process also acting in the common good. The consequences of a security breach are so large that this volunteer proactive activity should not only be encouraged but also rewarded and some companies are being paid handsomely for doing this as a business.

At present the tactical objective is to stay one step ahead of the crackers. We must think more strategically for the future. Social behavior, as it relates to computers and information technology, goes beyond merely adhering to the law since the law often lags technological advance. The physical activity of ethical hacking is sometimes hard to differentiate from cracking – it is hard to discern intent and predict future action – the main difference is that while an ethical hacker identifies vulnerabilities (often using the same scanning tools as a cracker) the ethical hacker does not exploit the vulnerabilities while a cracker does. Until a social framework is developed to discern the good guys (white hats) from the bad guys (black hats), we should be slow to codify into law or condemn ethical hacking – or we may risk eliminating our last thin line of stabilizing defense and not realize it until it is too late.

8. References

[1] B.J. Baird, L.L. Baird Jr., and R.P. Ranauro, "The Moral Cracker?" *Computers & Security*, No 6, 1987, pp. 471-478.

[2] M. Bishop, "Port Scanning," *Information Security Awareness Forum*, Department of Information Technology, State of California, Dec. 1 2000.

[3] G. Guissanie, "Information Assurance Red Teaming," *IAnewsletter*, Vol 2 No 4, Spring 1999, pp. 6, 8.

[4] J.A.N. Lee, "Hacking," *Macmillan Encyclopedia of Computers*, 1991.

[5] R. Linde, "Operating System Penetration," *National Computer Conference*, Vol 44 AFIPS Press, 1975.

[6] C.C. Palmer, "Ethical Hacking," *IBM Systems Journal*, Vol 40 No 3, 2001, pp. 769-780.

[7] M. Ranum, "The Network Police Blotter," *USENIX ;login:* August 2000, pp. 21-24.

[8] A. Ricadela, "The State of Software: Quality," *InformationWeek*, May 21 2001, p. 43.

[9] B. Smith, W. Yurcik, and D. Doss, "Ethical Hacking: The Security Justification," *Proceedings of the Ethics of Electronic Information in the 21st Century Symposium (EEI21)*, 2001.

[10] E.H. Spafford, "Are Computer Hacker Break-Ins Ethical?" *Journal of Systems Software, No 17*, 1992, pp. 41-47.

[11] J. Wack and M. Tracey, "DRAFT Guideline on Network Security Testing: Recommendations of the National Institute of Standards and Technology," *NIST Special Publication 800-42*, 2001.

[12] N. Wingfield, "It Takes a Hacker," *Wall Street Journal*, March 11 2002.

[13] B. Wood and R. Duggan, "Red Teaming of Advanced Information Assurance Concepts," *DARPA Information Survivability Conference & Exposition (DISCEX) – Volume II*, January 2000, pp. 112-118.

[14] W. Yurcik, A. Sharma, and D. Doss, "False Impressions: Contrasting Perceptions of Security as a Major Impediment to Achieving Survivable Systems," *IEEE/CERT/SEI 4th Information Survivability Workshop (ISW-2002)*, IEEE Computer Society Press, Vancouver, BC Canada, March 2002.