

Chapter 1

Introduction to Ethical Hacking

In This Chapter

- ▶ Understanding hacker and rogue insider objectives
 - ▶ Outlining the differences between ethical hackers and malicious attackers
 - ▶ Examining how the ethical hacking process has come about
 - ▶ Understanding the dangers that your computer systems face
 - ▶ Starting the ethical hacking process
-

This book is about hacking ethically — the science of testing your computers and networks for security vulnerabilities and plugging the holes you find before the bad guys get a chance to exploit them.

Although *ethical* is an often overused and misunderstood word, *Webster's New World Dictionary* defines *ethical* perfectly for the context of this book and the professional security testing techniques that I cover — that is, “conforming to the standards of conduct of a given profession or group.” IT practitioners are obligated to perform all the tests covered in this book above-board and only after permission has been obtained by the owner(s) of the systems — hence the disclaimer in the introduction.

Straightening Out the Terminology

We've all heard of external hackers and rogue insiders. Many of us have even suffered the consequences of their criminal actions. So who are these people? And why is it important to know about them? The next few sections give you the lowdown on malicious attackers.



In this book, I use the following terminology:

- ✓ *Hackers (or external attackers)* try to compromise computers and sensitive information for ill-gotten gains — usually from the outside — as an unauthorized user. Hackers go for almost any system they think they can compromise. Some prefer prestigious, well-protected systems, but hacking into anyone’s system increases their status in hacker circles.
- ✓ *Rogue insiders (or internal attackers)* try to compromise computers and sensitive information from the inside as authorized users. Rogue insiders go for systems they believe can be compromised for ill-gotten gains or revenge.

Malicious attackers are, generally speaking, both hackers and rogue insiders. For the sake of simplicity, I refer to both as *hackers* and specify *hacker* or *rogue insider* only when I need to drill down further into their tools, techniques, and ways of thinking.
- ✓ *Ethical hackers (or good guys)* hack a system to discover vulnerabilities for the purpose of protecting computers against illicit entry, abuse, and misuse.

Defining hacker

Hacker has two meanings:

- ✓ Traditionally, a hacker is someone who likes to tinker with software or electronic systems. Hackers enjoy exploring and learning how computer systems operate. They love discovering new ways to work — both mechanically and electronically.
- ✓ In recent years, *hacker* has taken on a new meaning — someone who maliciously breaks into systems for personal gain. Technically, these criminals are *crackers* (criminal hackers). Crackers break into (crack) systems with malicious intent. They are out for personal gain: fame, profit, and even revenge. They modify, delete, and steal critical information, often making other people miserable.

The good-guy (*white-hat*) hackers don’t like being in the same category as the bad-guy (*black-hat*) hackers. (In case you’re curious, the white-hat and black-hat terms come from Westerns in which the good guys wore white cowboy hats and the bad guys wore black cowboy hats.) There are also *gray-hat* hackers that are a little bit of both. Whatever the case, most people give *hacker* a negative connotation.

Many malicious hackers claim that they don't cause damage but instead are altruistically helping others. Yeah, right. Many malicious hackers are electronic thieves.

Defining rogue insider

Rogue insider — meaning a malicious employee, intern, or other user who abuses his or her privileges — is a term heard more and more within security circles and headlines talking about information breaches. An old statistic states that 80% of all security breaches are carried out by insiders. Whether or not this number is accurate is still questionable, but based on what I've seen and based on numerous annual surveys, there's undoubtedly an insider problem.

The issue is not necessarily users "hacking" internal systems, but rather users — from regular employees to auditors to contractors — who abuse the computer access privileges they've been given. There are cases of users ferreting through critical database systems to glean sensitive information, e-mailing confidential client information to the competition or other third parties, or deleting sensitive files from servers that they probably shouldn't have had access to in the first place. There's also the occasional "idiot insider" who's intent is not malicious but who still causes security problems nonetheless by moving, deleting, or otherwise corrupting sensitive information.

These rogue insiders are often our very worst enemies because they know exactly where to go to get the goods and don't need to be very computer-savvy in order to compromise very sensitive information.

How Malicious Attackers Beget Ethical Hackers

You need protection from hacker shenanigans; you need (or need to become) an ethical hacker. An *ethical hacker* possesses the skills, mindset, and tools of a hacker but is also trustworthy. Ethical hackers perform the hacks as security tests for their systems based on how a hacker or rogue insider would work.



Ethical hacking — which encompasses formal and methodical *penetration testing*, *white-hat hacking*, and *vulnerability testing* — involves the same tools, tricks, and techniques that hackers use, but with one major difference: Ethical hacking is legal because it's performed with the target's permission.

The intent of ethical hacking is to discover vulnerabilities from a malicious attacker's viewpoint so systems can be better secured. It's part of an overall information risk management program that allows for ongoing security improvements. Ethical hacking can also ensure that vendors' claims about the security of their products are legitimate.



If you perform ethical hacking tests for clients or simply want to add another certification to your credentials, you may want to consider becoming a Certified Ethical Hacker, a certification program sponsored by EC-Council. See www.eccouncil.org/CEH.htm for more information.

Understanding the Need to Hack Your Own Systems

To catch a thief, you must think like a thief. That's the basis for ethical hacking. It's absolutely critical to know your enemy. See Chapter 2 for details about how malicious attackers work.

The law of averages works against security. With the increased number and expanding knowledge of hackers, combined with the growing number of system vulnerabilities and other unknowns, the time will come when all computer systems are hacked or compromised in some way. Protecting your systems from the bad guys — and not just the generic vulnerabilities that everyone knows about — is absolutely critical. When you know hacker tricks, you can find out how vulnerable your systems really are.

Hacking preys on weak security practices and undisclosed vulnerabilities. Firewalls, encryption, and virtual private networks (VPNs) can create a false feeling of safety. These security systems often focus on high-level vulnerabilities, such as viruses and traffic through a firewall, without affecting how hackers work. Attacking your own systems to discover vulnerabilities is a big step toward making them more secure. This is the only proven method of greatly hardening your systems from attack. If you don't identify weaknesses, it's a matter of time before the vulnerabilities are exploited.

As hackers expand their knowledge, so should you. You must think like them and work like them in order to protect your systems from them. You, as the ethical hacker, must know the activities that hackers carry out and how to stop their efforts. You should know what to look for and how to use that information to thwart hackers' efforts.



You don't have to protect your systems from *everything*. You can't. The only protection against everything is to unplug your computer systems and lock them away so no one can touch them — not even you. That's not the best approach to information security and is certainly not good for business. What's important is to protect your systems from known vulnerabilities and common attacks.

It's impossible to *anticipate* all the possible vulnerabilities you'll have in your systems and business processes. You certainly can't plan for all possible attacks — especially the ones that are currently unknown. However, the more combinations you try — the more you test whole systems instead of individual units — the better your chances of discovering vulnerabilities that affect your information systems in their entirety.

Don't take ethical hacking too far, though. It makes little sense to harden your systems from unlikely attacks. For instance, if you don't have a lot of foot traffic in your office and no internal Web server running, you may not have as much to worry about as an Internet hosting provider would have. Your overall goals as an ethical hacker should be as follows:

- ✓ Hack your systems in a nondestructive fashion.
- ✓ Enumerate vulnerabilities and, if necessary, prove to management that vulnerabilities exist and can be exploited.
- ✓ Apply results to remove the vulnerabilities and better secure your systems.

Understanding the Dangers Your Systems Face

It's one thing to know that your systems generally are under fire from hackers around the world and rogue insiders around the office; it's another to understand specific attacks against your systems that are possible. This section offers some well-known attacks but is by no means a comprehensive listing.

Many information-security vulnerabilities aren't critical by themselves. However, exploiting several vulnerabilities at the same time can take its toll. For example, a default Windows OS configuration, a weak SQL Server administrator password, and a server hosted on a wireless network may not be major security concerns separately. But exploiting all three of these vulnerabilities at the same time can be a serious issue that leads to sensitive information disclosure and more.

Nontechnical attacks

Exploits that involve manipulating people — end users and even yourself — are the greatest vulnerability within any computer or network infrastructure. Humans are trusting by nature, which can lead to social-engineering exploits. *Social engineering* is the exploitation of the trusting nature of human beings to gain information for malicious purposes.

Other common and effective attacks against information systems are physical. Hackers break into buildings, computer rooms, or other areas containing critical information or property to steal computers, servers, and other valuable equipment. Physical attacks can also include *dumpster diving* — rummaging through trash cans and dumpsters for intellectual property, passwords, network diagrams, and other information.

Network infrastructure attacks

Hacker attacks against network infrastructures can be easy because many networks can be reached from anywhere in the world via the Internet. Here are some examples of network-infrastructure attacks:

- ✔ Connecting into a network through a rogue modem attached to a computer behind a firewall
- ✔ Exploiting weaknesses in network protocols, such as TCP/IP and NetBEUI
- ✔ Flooding a network with too many requests, creating a denial of service (DoS) for legitimate requests
- ✔ Installing a network analyzer on a network and capturing every packet that travels across it, revealing confidential information in clear text
- ✔ Piggybacking onto a network through an unsecure 802.11 wireless configuration

Operating system attacks

Hacking operating systems (OSes) is a preferred method of the bad guys. OS attacks make up a large portion of hacker attacks simply because every computer has one and so many well-known exploits can be used against them.

Occasionally, some operating systems that appear to be more secure out of the box — such as Novell NetWare and various flavors of BSD UNIX — are attacked, and vulnerabilities turn up. But hackers often prefer attacking operating systems such as Windows and Linux because they are widely used and better known for their publicized weaknesses.

Here are some examples of attacks on operating systems:

- ✓ Exploiting specific network protocol implementations
- ✓ Attacking built-in authentication systems
- ✓ Breaking file system security
- ✓ Cracking passwords and encryption mechanisms

Application and other specialized attacks

Applications take a lot of hits by hackers. Programs such as e-mail server software and Web applications are often beaten down:

- ✓ Hypertext Transfer Protocol (HTTP) and Simple Mail Transfer Protocol (SMTP) applications are frequently attacked because most firewalls and other security mechanisms are configured to allow full access to these services from the Internet.
- ✓ Voice over IP (VoIP) faces increasing attacks as it finds its way into more and more businesses.
- ✓ Unsecure files containing sensitive information are scattered throughout workstation and server shares, and database systems contain numerous vulnerabilities — all of which can be exploited by rogue insiders.

Ethical hacking helps carry out such attacks against your computer systems and highlights any associated weaknesses. Parts II through V of this book cover these attacks in detail, along with specific countermeasures you can implement against attacks on your systems.

Obeying the Ethical Hacking Commandments

Every ethical hacker must abide by a few basic commandments. If not, bad things can happen. I've seen these commandments ignored or forgotten when

planning or executing ethical hacking tests. The results weren't positive — trust me.

Working ethically

The word *ethical* in this context can be defined as working with high professional morals and principles. Whether you're performing ethical hacking tests against your own systems or for someone who has hired you, everything you do as an ethical hacker must be aboveboard and must support the company's goals. No hidden agendas are allowed!

Trustworthiness is the ultimate tenet. The misuse of information is absolutely forbidden. That's what the bad guys do. Let them be the ones who get fined or go to prison because of their bad choices.

Respecting privacy

Treat the information you gather with the utmost respect. All information you obtain during your testing — from Web application log files to clear text passwords — must be kept private. Don't snoop into confidential corporate information or employees' private lives. If you sense that privacy is being breached by a colleague or team member and you feel like someone should know about it, consider sharing that information with the appropriate manager.



Involve others in your process. This is a “watch the watcher” system that can build trust and support for your ethical hacking projects.

Not crashing your systems

One of the biggest mistakes I've seen when people try to hack their own systems is inadvertently crashing the very systems they're trying to keep running. The main reason for this is poor planning. These testers have not read the documentation or misunderstand the usage and power of the security tools and techniques.

You can easily create DoS conditions on your systems when testing. Running too many tests too quickly can cause system lockups, data corruption, reboots, and more. I know because I've done this! Don't rush things and assume that a network or specific host can handle the beating that network scanners and vulnerability assessment tools can dish out.



Many security assessment tools can control how many tests are performed on a system at the same time. These tools are especially handy if you need to run the tests on production systems during regular business hours.

You can even accidentally create an account or system lockout condition by socially engineering someone into changing a password, not realizing that doing so might create a system lockout condition.

The Ethical Hacking Process

Like practically any IT or security project, ethical hacking needs to be planned in advance. Strategic and tactical issues in the ethical hacking process should be determined and agreed upon. To ensure the success of your efforts, spend time up front planning things out. Planning is important for any amount of testing — from a simple password-cracking test to an all-out penetration test on a Web application.



If you choose to hire a “reformed” hacker to work with you during your testing or to obtain an independent perspective, there are many things you must consider. I cover the pros and cons and do’s and don’ts associated with hiring an ethical hacker in Chapter 18.

Formulating your plan

Approval for ethical hacking is essential. Make what you’re doing known and visible — at least to the decision makers. Obtaining *sponsorship* of the project is the first step. This could be your manager, an executive, your client, or even yourself if you’re the boss. You need someone to back you up and sign off on your plan. Otherwise, your testing may be called off unexpectedly if someone claims they never authorized you to perform the tests.

The authorization can be as simple as an internal memo or e-mail from your boss if you’re performing these tests on your own systems. If you’re testing for a client, have a signed contract in place, stating the client’s support and authorization. Get written approval on this sponsorship as soon as possible to ensure that none of your time or effort is wasted. This documentation is your *Get Out of Jail Free* card if anyone questions what you’re doing, or worse, if the authorities come calling.

One slip can crash your systems — not necessarily what anyone wants. You need a detailed plan, but that doesn’t mean you need volumes of testing procedures. A well-defined scope includes the following information:

✔ **Specific systems to be tested:** When selecting systems to test, start with the most critical systems and processes or the ones you suspect to be the most vulnerable. For instance, you can test computer passwords, an Internet-facing Web application, or attempt social engineering attacks before drilling down into all your systems.

✔ **Risks involved:** It pays to have a contingency plan for your ethical hacking process in case something goes awry. What if you're assessing your firewall or Web application and you take it down? This can cause system unavailability, which can reduce system performance or employee productivity. Even worse, it could cause loss of data integrity, loss of data itself, and even bad publicity. It'll most certainly tick off a person or two and make you look bad.

Handle social engineering and DoS attacks carefully. Determine how they can affect the systems you're testing and your entire organization.

✔ **When the tests will be performed and your overall timeline:** Determining when the tests are performed is something that you must think long and hard about. Do you perform tests during normal business hours? How about late at night or early in the morning so that production systems aren't affected? Involve others to make sure they approve of your timing.

The best approach is an unlimited attack, wherein any type of test is possible at any time of day. The bad guys aren't breaking into your systems within a limited scope, so why should you? Some exceptions to this approach are performing DoS attacks, social engineering, and physical security tests.

✔ **How much knowledge of the systems you have before you start testing:** You don't need extensive knowledge of the systems you're testing — just a basic understanding. This basic understanding helps protect you and the tested systems.

Understanding the systems you're testing shouldn't be difficult if you're hacking your own in-house systems. If you're testing a client's systems, you may have to dig deeper. In fact, I've never had a client ask for a fully blind assessment. Most IT managers and others responsible for security are scared of these assessments — and they can take more time and cost more to boot. Base the type of test you will perform on your organization's or client's needs.

✔ **What action will be taken when a major vulnerability is discovered:** Don't stop after you find one security hole. This can lead to a false sense of security. Keep going to see what else you can discover. I'm not saying to keep hacking until the end of time or until you crash all your systems; simply pursue the path you're going down until you can't hack it any longer (pun intended). If you haven't found any vulnerabilities, you haven't looked hard enough.

- ✔ **The specific deliverables:** This includes security assessment reports and a higher-level report outlining the general vulnerabilities to be addressed, along with countermeasures that should be implemented.

One of your goals may be to perform the tests without being detected. For example, you may be performing your tests on remote systems or on a remote office, and you don't want the users to be aware of what you're doing. Otherwise, the users may catch on to you and be on their best behavior — instead of their normal behavior.

Selecting tools

As with any project, if you don't have the right tools for ethical hacking, accomplishing the task effectively is difficult. Having said that, just because you use the right tools doesn't mean that you will discover all vulnerabilities.



Know the personal and technical limitations. Many security assessment tools generate false positives and negatives (incorrectly identifying vulnerabilities). Others just skip right over vulnerabilities altogether. If you're performing tests such as social engineering or physical security assessments, you may miss weaknesses because security testing tools aren't quite that smart.

Many tools focus on specific tests, and no tool can test for everything. For the same reason you wouldn't drive in a nail with a screwdriver, you shouldn't use a word processor to scan your network for open ports. This is why you need a set of specific tools that you can call on for the task at hand. The more (and better) tools you have, the easier your ethical hacking efforts are.

Make sure you're using the right tool for the task:



- ✔ To crack passwords, you need cracking tools like `pwdump3` and Proactive Password Auditor.
A general port scanner, such as SuperScan or Nmap, just won't work for cracking passwords.
- ✔ For an in-depth analysis of a Web application, a Web application assessment tool (such as N-Stalker or WebInspect) is more appropriate than a network analyzer (such as Ethereal).



When selecting the right security tool for the task, ask around. Get advice from your colleagues and from other people online. A simple groups search on Google (<http://groups.google.com>) or perusal of security portals, such as <http://SecurityFocus.com>, <http://SearchSecurity.com>, and www.ITsecurity.com, often produces great feedback from other security experts.

Hundreds, if not thousands, of tools can be used for ethical hacking — from your own words and actions to software-based vulnerability assessment programs to hardware-based network analyzers. The following list runs down some of my favorite commercial, freeware, and open source security tools:

- ✓ Cain and Abel
- ✓ EtherPeek
- ✓ SuperScan
- ✓ QualysGuard
- ✓ WebInspect
- ✓ Proactive Password Auditor
- ✓ LANguard Network Security Scanner
- ✓ RFprotect Mobile
- ✓ ToneLoc

I discuss these tools and many others in Parts II through V when I go into the specific hack attacks. Appendix A contains a more comprehensive listing of these tools for your reference.

The capabilities of many security and hacking tools are often misunderstood. This misunderstanding has cast negative light on otherwise excellent and legitimate tools.

Some of these security testing tools are complex. Whichever tools you use, familiarize yourself with them before you start using them. Here are ways to do that:

- ✓ Read the readme and/or online help files for your tools.
- ✓ Study the user's guides for your commercial tools.
- ✓ Use the tools in a lab/test environment.
- ✓ Consider formal classroom training from the security-tool vendor or another third-party training provider, if available.

Look for these characteristics in tools for ethical hacking:

- ✓ Adequate documentation
- ✓ Detailed reports on the discovered vulnerabilities, including how they may be exploited and fixed
- ✓ General industry acceptance

- ✓ Availability of updates and support
- ✓ High-level reports that can be presented to managers or nontechnie types

These features can save you a ton of time and effort when you're performing your tests and writing your final reports.

Executing the plan

Good ethical hacking takes persistence. Time and patience are important. Be careful when you're performing your ethical hacking tests. A hacker in your network or a seemingly benign employee looking over your shoulder may watch what's going on and use this information against you.

It isn't practical to make sure that no hackers are on your systems before you start. Just make sure you keep everything as quiet and private as possible. This is especially critical when transmitting and storing your test results. If possible, encrypt any e-mails and files containing sensitive test information by using Pretty Good Privacy (PGP) (www.pgp.com) or similar technology. At a minimum, password-protect them.

You're now on a reconnaissance mission. Harness as much information as possible about your organization and systems, which is what malicious hackers do. Start with a broad view and narrow your focus:

- 1. Search the Internet for your organization's name, your computer and network system names, and your IP addresses.**

Google is a great place to start.

- 2. Narrow your scope, targeting the specific systems you're testing.**

Whether you're assessing physical security structures or Web applications, a casual assessment can turn up a lot of information about your systems.

- 3. Further narrow your focus with a more critical eye. Perform actual scans and other detailed tests to uncover vulnerabilities on your systems.**

- 4. Perform the attacks and exploit any vulnerabilities you've found, if that's what you choose to do.**

Evaluating results

Assess your results to see what you uncovered, assuming that the vulnerabilities haven't been made obvious before now. This is where knowledge counts. Evaluating the results and correlating the specific vulnerabilities discovered is a skill that gets better with experience. You'll end up knowing your systems much better than anyone else. This makes the evaluation process much simpler moving forward.



Submit a formal report to upper management or to your client, outlining your results and any recommendations you wish to share. Keep these parties in the loop to show that your efforts and their money are well spent. Chapter 16 describes the ethical hacking reporting process.

Moving on

When you've finished your ethical hacking tests, you (or your client) still need to implement your recommendations to make sure the systems are secure.



New security vulnerabilities continually appear. Information systems constantly change and become more complex. New hacker exploits and security vulnerabilities are regularly uncovered. You may discover new ones! Security tests are a snapshot of the security posture of your systems. At any time, everything can change, especially after upgrading software, adding computer systems, or applying patches. Plan on testing regularly and consistently (for example, once a month, once a quarter, or bi-annually). Chapter 18 covers managing security changes.