8 Legal Issues

Richard Salgado

The views expressed in this chapter are those of Richard Salgado and do not necessarily represent the views of the Department of Justice.

Identifying the right technical configurations is only part of the job of designing and deploying a honeynet. There are also legal issues that you need to consider to reduce the risk that you will find yourself embroiled in litigation or otherwise entangled with the legal system. Failing to address these properly can make your honeynet an expensive liability. Forethought can go a long way to avoiding these legal pitfalls, and with an understanding of the relevant laws, you can take steps to reduce your exposure.

In this chapter, I will first address the limitations imposed on network operators who would like to monitor the activities of system users. The law in this area is developing, and there are discernible rules that may be surprising to lawyers and nonlawyers alike. Second, I address the possibility that your honeynet will detect improper activity, discuss what types of conduct are criminal in the U.S., and describe protocols that may be helpful in the event your honeynet becomes a witness to a crime. Third, I discuss the possibility of liability for running a honeynet that injures others.

The bottom line for the entire discussion is that you should consult with your lawyer before you design or deploy your honeynet. If you are considering a honeynet for your organization, check with counsel who advises the organization. In the case of a large enterprise, there may be in-house counsel who can provide the necessary guidance; if not, your enterprise may need to consult with outside counsel. For government agencies, there may be an office of general counsel, Inspector General, or other source of advice. (Government organizations in the U.S. may also consult with the Computer Crime and Intellectual Property Section in the Department of Justice for guidance.) Your counsel will take into account your particular situation and goals, the regulations, state law, and local law applicable to you, and will help you identify potential problems and solutions.

Many of the concerns I discuss here apply equally to computer networks generally, even those that are not honeynets.

MONITORING NETWORK USERS

The first point is one that often surprises many people: Just because you own and are responsible for a computer network does not mean that you have unfettered legal authority to monitor users of the network, even if your network is a honeynet populated exclusively by intruders. There are many possible sources of restrictions that could make monitoring improper (such as statutes, internal policies, and user agreements). Failing to honor these restrictions could land you in civil and even criminal hot water. In the honeynet context, these rules take on particular significance because the entire value of the honeynet may be tied to monitoring. I first address the potential restrictions found in the U.S. Constitution and federal statutes.

U.S. CONSTITUTIONAL PROVISIONS

If your honeynet is operated at the direction of the government, consider the (unlikely) possibility that the Fourth Amendment to the U.S. Constitution could apply. The Fourth Amendment limits the power of government agents to search for evidence without having first secured a search warrant from a judge. Evidence seized in violation of the Fourth Amendment may not be admissible at a criminal

trial against the person who was subjected to the illegal search. In addition, the person who violated the Fourth Amendment rights of another may be subject to a lawsuit for money damages.

The Fourth Amendment applies only where the person searched has a "reasonable expectation of privacy." Those who hack into networks do not have a "reasonable" expectation of privacy in their use of the victim network.[1] In addition, the Fourth Amendment restricts searches only by the government; a private actor may deploy a honeynet and monitor users without worrying about the Fourth Amendment, unless the private actor is an instrument or agent of the government.[2] Similar provisions in state constitutions are at least as rigorous as the federal Constitution, and perhaps more.

Think about whether your organization is subject to the Fourth Amendment; you might be surprised to discover that your organization is a government entity for the purpose of the amendment. For example, because of their research value, academics and students may be drawn to the idea of deploying honeynets with an eye toward studying the results. If the honeynet is deployed in connection with a *public* university, the rules of the Fourth Amendment may well apply to the monitoring. Of course, as I noted above, a honeynet that monitors only the activities of intruders will not violate the Fourth Amendment because intruders do not have a reasonable expectation of privacy. If the scope of the monitoring goes beyond the intruders, however, the Fourth Amendment issue may be very real.

U.S. STATUTES

In the U.S., there are privacy laws that can apply to the operation of a honeynet. The two federal statutes most worthy of discussion here are the Wiretap Act and the awkwardly named Pen Register, Trap and Trace Devices statute. The Wiretap

1. *U.S. v. Seidlitz,* 589 F.2d 152, 160 (4th Cir. 1978) ("having been 'caught with his hand in the cookie jar,'" hacker has no constitutional right to suppression of evidence gathered from victim computer); see *Rakas v. Illinois,* 439 U.S. 128, 143 n.12 (1978) (burglar has no reasonable expectation of privacy while on victim premises); *Compuserve, Inc. v. Cyber Promotions, Inc.,* 962 F. Supp. 1015, 1021 (S.D. Ohio 1997) (courts have likened computer hacking and trespassing).
2. *U.S. v. Jacobson,* 466 U.S. 109, 115 (1984).

Act covers the interception of the contents of communications. The Pen Register, Trap and Trace Devices statute covers the collection of noncontent information. This is not an easy area of the law, but the penalties for violation can be severe. Do not ignore these rules.

The Wiretap Act

The federal Wiretap Act generally forbids the interception of the content of communications (including electronic communications) unless one of the exceptions listed in the statute applies. Sniffing traffic on a network may be considered an interception of electronic communications and would fall within the scope of the Wiretap Act.[3] A violation of the Wiretap Act is no small matter. It can lead to a civil suit and may constitute a federal felony punishable by a fine and up to five years in prison.[4]

If your honeynet is not configured to capture the content of communications of users, then there is no Wiretap Act issue.[5] Thus, for example, if you operate a low-interaction honeynet, you may have it configured to log only the IP addresses and port calls of incoming connection attempts. If so, then the honeynet is acquiring only communications-related data, but not the content of any communications themselves. The Wiretap Act would not apply (although the Pen Register, Trap and Trace Devices statute may).

---

3. In re *Pharmatrak, Inc. Privacy Litig'n,* No. CIV.A.00-11672-JLT, 2002 WL 1880387 (D. Mass., Aug. 13, 2002); In re *DoubleClick Inc. Privacy Litig'n,* 154 F.Supp.2d 497 (S.D.N.Y. 2001).
4. 18 U.S.C. § 2511(4) & (5).
5. In the course of running a honeynet, you may find that users are uploading data to the honeynet. An intruder may, for example, set up file transfer protocol ("FTP") services and store files for later retrieval. Looking at those stored files probably would not implicate the Wiretap Act, because there would be no interception of the communications in transit. Accessing the stored communications of users may implicate the stored communication portion of the Electronic Communications Privacy Act (ECPA). ECPA creates privacy rights for customers and subscribers of certain computer network service providers. 18 U.S.C. §§ 2701–2712. For a comprehensive, but accessible, discussion of those rules, see "Searching and Seizing Computers and Obtaining Electronic Evidence in Criminal Investigations" published by the Computer Crime and Intellectual Property Section of the U.S. Department of Justice (*http://www.cybercrime.gov/searching.html*).

As a constitutional matter, an intruder has no reasonable expectation of privacy while in your network. This does not mean, however, that monitoring is allowed under the Wiretap Act. Strange as it may seem, a hacker may have no reasonable expectation of privacy under the Constitution, but may nonetheless have privacy rights under the Wiretap Act.

The Wiretap Act contains many exceptions to the prohibition against intercepting the contents of communications. With regard to honeynets and other computer systems, exceptions to consider include the "provider protection" exception and the "consent of a party" exception. If monitoring is done by the government, the "computer trespasser" exception may also apply.

The Provider Protection Exception The "provider protection" exception allows an electronic communication service provider to intercept communications to protect the provider's rights or property.[6] Providers can monitor communications over their system to prevent, for example, abuse or damage to the system. This exception allows network operators to monitor hostile activity, run intrusion detection software, and scan the contents of inbound traffic for malcode signatures without violating the Wiretap Act.

The exception states:

> It shall not be unlawful under [the Wiretap Act] for an operator of a switchboard, or an officer, employee, or agent of a provider of wire or electronic communication service, whose facilities are used in the transmission of a wire or electronic communication, to intercept, disclose, or use that communication in the normal course of his employment while engaged in any activity which is a necessary incident to the rendition of his service or to the protection of the rights or property of the provider of that service, except that a provider of wire communication service to the public shall not utilize service observing or random monitoring except for mechanical or service quality control checks.[7]

6. 18 U.S.C. § 2511(2)(a)(i).
7. 18 U.S.C. § 2511(2)(a)(i).

Under this exception, providers may listen and record communications to prevent against fraud, theft of services, damage, and privacy invasions, for example. Even if the monitoring is done to assist law enforcement to pursue a criminal investigation, the exception is proper if it serves to protect the provider's rights or property.[8]

This is not an unlimited exception, however. Providers must balance the need to protect their rights and property with the privacy needs of the legitimate users of the services. Monitoring is permitted under the Wiretap Act if there is a "substantial nexus" between the monitoring done and the threat to the provider's rights or property.[9] Where the monitoring is done for other purposes, it will fall outside the exception and may violate the Wiretap Act if no other exception applies. This was the situation a cellular phone provider found itself in when assisting the police to investigate a kidnapper. The kidnapper had made calls from a cloned cell phone, and the police asked the cell phone company to intercept communications in the hopes of learning who the kidnapper was, and finding the victim. The company agreed and listened to calls to and from the cloned phone. From the intercepted calls, the police were able to find and arrest the kidnapper. Amazingly, the kidnapper then sued the police for violating the Wiretap Act, arguing that the provider protection exception did not apply to the interception. The trial court agreed. The court found that the exception did not apply because the phone company was not intercepting to protect its rights or property (for example, preventing theft of services); it was done to advance the interest of the police in investigating the kidnapping.[10]

The courts have not addressed whether the provider protection exception applies to interceptions of communications to or from a honeynet. There is some tension between the claim that sniffing traffic on a honeynet is done to protect the rights or property of the honeynet operator and the fact that the honeynet is deployed for the very purpose of being attacked. Arguably, sniffing on a honeynet

8. See *U.S. v. Harvey,* 540 F.2d 1345, 1352 (8th Cir. 1976).
9. See *U.S. v. Auler,* 539 F.2d 642, 646 (7th Cir. 1976) (telephone company); *United States v. Harvey,* 540 F.2d 1345, 1350 (8th Cir. 1976); *United States v. Freeman,* 524 F.2d 337, 340 (7th Cir. 1975); *United States v. McLaren,* 957 F. Supp. 215, 219 (M.D. Fla. 1997).
10. *McClelland v. McGrath,* 31 F. Supp. 2d 616 (N.D. Ill. 1998).

is not done by the provider to protect the honeynet; rather, the honeynet is there to give the provider something to sniff.

This is not to say that the provider protection exception would never apply; the courts simply have not yet addressed the issue, and there is a risk that the courts will reject its application. So how can you deal with this risk? First, certain honeynet configurations may give you a better argument that the exception applies than do other configurations. By carefully planning your configuration, you may be able to strengthen your argument that the honeynet has a role in protecting other parts of your network. (I address examples of such configurations below.) Second, whatever configuration you ultimately deploy, take the time to document the protective purposes of the honeynet. If called on later to prove that the exception applies, you will find that documentation very useful to support your argument that the purpose of the honeynet was to protect other servers.

The bigger the role a honeynet plays in protecting a production server or network, the better the chance that the provider protection exception will apply. Below are five examples of honeynets, each of which may be viewed differently by a court based on its value in protecting a production server.

- **Example 1: Independent from Production Server** In this scenario, the honeynet is unrelated to any particular production server, as shown in Figure 8-1. The most that can be said about the protective value of this honeynet is that it may, in a long-term and somewhat abstract sense, protect production servers across the Internet by generally enhancing the art of network attack detection, prevention, and response.

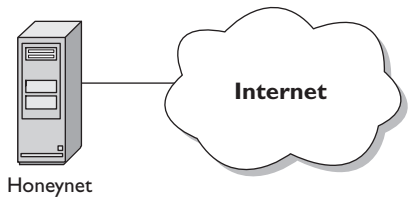

Figure 8-1 This honeynet is unrelated to any particular production server.

■ **Example 2: Configuration-Dependent** In this scenario, the honeynet is configured identically, in material respects, to a particular production server of the honeynet operator, as shown in Figure 8-2. It may run the same operating system, use the same hardware, run the same services, use the same firewall and signatures, or be identical in other ways to a particular production server. The honeynet is not, however, connected in the same subnet as the production server. The goal of this honeynet may be to secure a production server (or class of production servers) operated by the honeynet owner by (a) revealing the attacks (and perhaps identifying the signatures of the attacks) that are directed at the particular configuration, (b) revealing vulnerabilities that exist in that configuration, and (c) making it easier to develop and test response tactics to limit the effectiveness of the attacks.

If sued for a federal Wiretap Act violation for sniffing traffic on the honeynet, the operator of this honeynet may be able to argue that the provider protection exception allowed for the monitoring because it led to enhanced security measures for the operator's production servers. If the honeynet operator has documented that this was a goal of the honeynet from the outset, and has documented the security improvements implemented on the production servers that were developed as a result of lessons learned from the honeynet monitoring, the operator has increased the chance that a court will agree that the exception applies.

■ **Example 3: IP Address-Dependent** In this scenario, the honeynet is nestled within a contiguous IP address range used by production servers of the honeynet operator, as shown in Figure 8-3. Scans of the IP address range will cover the production servers as well as the honeynet. The honeynet can be configured

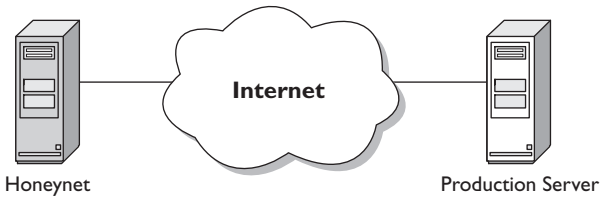

Figure 8-2 This honeynet is configured identically, in material respects, to a particular production server of the honeynet operator.

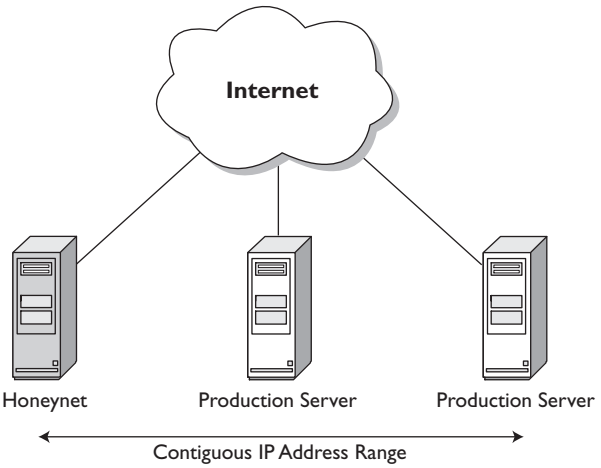

Figure 8-3 This honeynet is nestled within a contiguous IP address range used by production servers of the honeynet operator.

to appear identical, in material respects, to one or more of its production server neighbors on the network. The goal of the honeynet operator may be to secure his or her production neighbors by (a) revealing the attacks (and perhaps identifying the signatures of the attacks) that are directed at the configuration used by the production servers, (b) revealing vulnerabilities that exist in that configuration, and (c) making it easier to develop and test response tactics to limit the effectiveness of the attacks.

In addition, the honeynet could be configured to be vulnerable to particular types of attacks and populated with tantalizing data. The attention of a would-be attacker who scans the IP address block for vulnerable computers may be drawn to the honeynet and away from the relatively secure production servers. The honeynet protects the production servers, the operator may argue, by acting as a lightning rod for attacks that would have otherwise hit the production servers.

■ **Example 4: Demilitarized Zone** In this scenario, the honeynet is located behind a firewall in the so-called demilitarized zone (DMZ) with production servers such as mail or web servers, but separate from the rest of the internal network, as shown in Figure 8-4. The honeynet could be listening to the ports

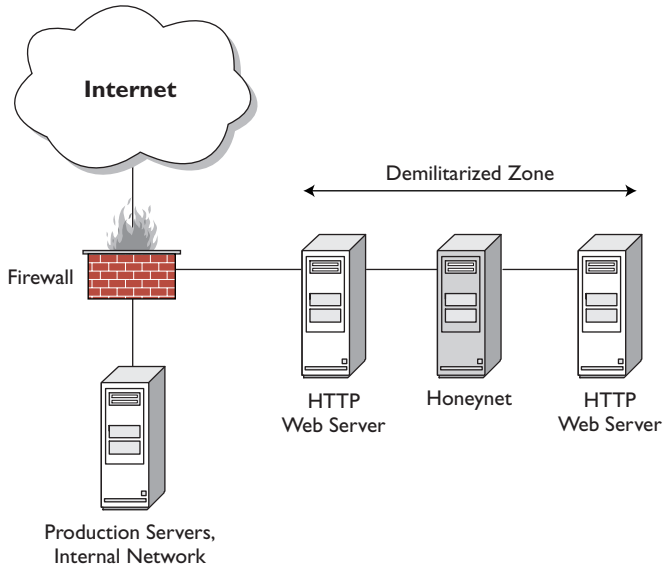

Figure 8-4 This honeynet is located behind a firewall in the so-called "DMZ" with production servers such as mail or web servers, but separate from the rest of the internal network.

served by the other servers in the DMZ. Any connections to those ports would be presumed attacks, and likely attacks that are also being launched against the other servers. Unlike the other servers, however, the honeynet may be taken down and analyzed without disrupting services offered to legitimate users. This enhances the organization's ability to identify attacks that are in all likelihood also being directed at the other DMZ servers, identifying vulnerabilities that exist in those servers, and finding means to prevent and respond to the attacks. The honeynet operator could argue that the honeynet protects the servers in the DMZ by acting as an attack lightning rod, and also serves to identify attacks that may be intended for the internal production servers.

■ **Example 5: Sandbox** In this scenario, the honeynet is actually part of the production server, as shown in Figure 8-5. It is also referred to as a "sandbox." The software Back Officer Friendly by NFR Security, Inc. is a simple example of this approach. The sandbox honeynet sees attacks or attempted attacks against the very production server on which the honeynet is running. The sandbox,

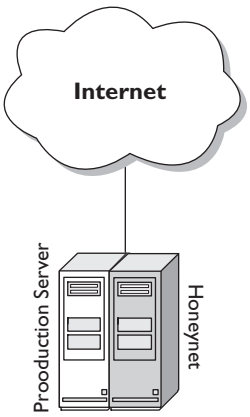

Figure 8-5 This honeynet is actually part of the production server. Such honeynets are also at times referred to as "sandboxes."

compared with the other types of honeynets, plays a relatively immediate role in the protection of the production server. The honeynet operator could argue that the honeynet, intimately associated with the production server, plays a direct role in preventing attacks against the production server.

The Consent of a Party Exception The "consent of a party" exception is fairly intuitive.[11] If a party to a communication has consented to monitoring (or if a party actually does the intercepting), the interception is permitted under the Wiretap Act (unless it was done for some other unlawful purpose). A honeynet operator may be able to get consent from attackers by placing a "consent banner" on the honeynet. The banner would tell users (including would-be attackers) that by accessing the system they are consenting to monitoring. If a hacker uses the system having seen the banner, the hacker has assented to the terms and given the system operator consent to monitor the session.

In addition, arguably when an intruder communicates with the honeynet (for example by uploading a file), the honeynet itself is a party to the communication

11. 18 U.S.C. § 2511(2)(c)–(d).

and can consent to monitoring.[12] This interpretation of the consent exception runs into difficulty, however, when the attacker uses the computer as a hop-through to connect with another computer. For example, if the attacker connects to a honeynet, then uses the bandwidth of the honeynet to connect with another victim computer, the honeynet stops looking so much like a party to the communication with the attacker and more like a switch between the attacker and the other victim. A honeynet operator could eliminate this possibility of being used as a hop-through by logging attempted outbound connections but blocking them and instead returning a failure reply to the attacker. There is a price to be paid by such a configuration: The honeynet may look less "real" and may be less attractive to the attackers of interest.

The Computer Trespasser Exception The "computer trespasser" exception, enacted as part of the USA PATRIOT Act, allows the government to monitor hackers in certain situations.[13] It applies where the user being monitored is a trespasser and the communications monitored are relevant to an ongoing investigation. Government must, of course, secure permission of the owner or operator of the network before monitoring. This exception may be useful for honeynet owners, particularly when the honeynet is run with a government entity.

The Pen Register, Trap and Trace Devices Statute

The Pen Register, Trap and Trace Devices statute (Pen/Trap statute) governs the real-time collection of non-content traffic information associated with communications, such as the phone numbers dialed by a particular telephone or the destination or source IP address of a computer network user (data the statute refers to as "dialing, routing, addressing, or signaling information").[14] Like the Wiretap Act's prohibition on interception of the contents of communications, the Pen/Trap statute creates a general prohibition on the real-time monitoring of traffic data relating to communications. A pen register is a device or process that records outgoing connection information (for example, the telephone number

12. *U.S. v. Mullins,* 992 F.2d 1472, 1478 (9th Cir. 1993); *U.S. v. Seidlitz,* 589 F.2d 152, 158 (4th Cir. 1978); *In re DoubleClick Inc. Privacy Litig'n,* 154 F.Supp.2d 497 (S.D.N.Y. 2001).
13. 18 U.S.C. § 2511(2)(i).
14. 18 U.S.C. §§ 3121–3127.

dialed from a monitored telephone); a trap and trace device captures incoming connection information (for example, the phone number of a call to the monitored telephone).

The Pen Register, Trap and Trace Devices statute generally forbids the acquisition of noncontent information of a communication, unless one of the listed exceptions applies. In the computer network context, this includes, for example, network routing information, such as the source and destination IP address, the port number that handled that communication, and email addresses of the attackers. If the device or process is intended to capture content of communications, such as the subject line or body of an email or the content of a downloaded file, then its use is governed by the Wiretap Act, not the Pen Register, Trap and Trace Devices statute.

Through the Pen Register, Trap and Trace Devices statute, Congress gives network operators plenty of authority to use Pen/Trap devices on their networks. The statute has never been tested in court, however, as applied to honeynets. Providers are permitted to use Pen/Trap devices as follows:

- Relating to the operation, maintenance, and testing of a wire or electronic communication service or to the protection of the rights or property of such provider, or to the protection of users of that service from abuse of service or unlawful use of service
- To record the fact that a wire or electronic communication was initiated or completed in order to protect such provider, another provider furnishing service toward the completion of the wire communication, or a user of that service, from fraudulent, unlawful, or abusive use of service
- Where the consent of the user of that service has been obtained[15]

Notice that these exceptions follow the exceptions in the Wiretap Act. Generally speaking, if a honeynet operator fits within one of the exceptions to the Wiretap Act for intercepting the contents of communications, the operator is also authorized to intercept the noncontent information concerning the communication.

15. 18 U.S.C. § 3121(b).

Like a Wiretap Act violation, violation of the Pen Register, Trap and Trace Devices statute is a crime. Violations are punishable by a fine and up to a year in prison.[16] Treat the matter seriously, and consult with counsel.

U.S. CONTRACTS AND POLICIES

Another source of privacy rights in the U.S. for users may be contract and policies. An organization that has promised users that it will not engage in certain types of network monitoring may find itself in a lawsuit if it breaks such a promise. If your honeynet is deployed on part of a network that is subject to such monitoring contracts or policies, take care that you take them into account.

LAWS OUTSIDE THE U.S.

Countries other than the U.S. also have laws that apply to the operation of honeynets. A complete catalog of all these laws of all the jurisdictions in all the countries is well beyond the scope of this chapter. Consult counsel who knows the laws in the jurisdiction in which you plan to deploy your honeynet.

CRIME AND THE HONEYNET

Intruders on your honeynet may have nasty plans for you. Not only may an attacker intend to victimize you, the attacker may want to use your system as a launching point to attack others with your bandwidth. The attacker may want to stash contraband, such as stolen credit cards, password files, or trade secrets; perhaps the attacker will try to set up a "warez" site to traffic in pirated software or entertainment media, or use your system to distribute child pornography. Don't let your honeynet become part of the problem. Below I discuss some of the types of illicit conduct that you may see on your honeynet that could form the basis for criminal or civil action against the attacker, and provide some ideas on how to deal with evidence you collect. Before you take your honeynet live, have a plan in place for dealing with criminal conduct that your honeynet may witness.

16. 18 U.S.C. § 3121(d).

COMMON TYPES OF CRIMINAL ACTIVITY

There is a myriad of conduct that you may see on your honeynet that constitutes or evinces one or more crimes under U.S. law and that could lead to a civil lawsuit against the attacker. The most obvious crime you may expect is a network intrusion (or attempted network intrusion). There are other crimes that you may detect in the course of operating a honeynet, however. I deal with only a few here. Again, it is a good idea to consult with an attorney before and while you operate your honeynet. For many of the crimes you may see, you will want to be able to respond quickly and responsibly.

Network Crimes

In the U.S., most of the computer network crimes are defined in the federal Computer Fraud and Abuse Act.[17] I concentrate on this statute, although there are others that can apply as well, depending on the facts. In addition, most states in the U.S. have computer-crime laws that criminalize unauthorized access or damage to a computer or network.[18] The laws of other countries vary widely, but many make intrusions and network attacks criminal.[19]

The Computer Fraud and Abuse Act criminalizes certain attacks against certain computers by certain actors. Although there are many sections and subsections in the statute that cover many different types of attacks, there are a set of provisions commonly applied to most network attacks. These provisions cover so-called "protected computers."

A "protected computer" includes any computer "used in interstate or foreign commerce or communication." Basically, any computer on the Internet is "protected" under the statute because it is used in interstate communication. In

17. The Computer Fraud and Abuse Act is found at 18 U.S.C. § 1030. The text of the statute and examples of cases prosecuted under that statute can be found at *http://www.cybercrime.gov*.
18. For a partial list of state computer crime laws see *http://nsi.org/Library/Compsec/computerlaw/statelaws.html*.
19. For a partial list of computer crime laws of jurisdictions outside the U.S. see *http://www.mossbyrett.of.no/info/legal.html*.

addition, all U.S. government computers, and those used by banks and other financial institutions, are considered "protected" under the statute.[20] Computers outside the U.S. can also be "protected" under the statute.[21] This means that it can be criminal for an attacker located in the U.S. to victimize a computer located outside the U.S. (It also allows U.S. law enforcement to provide faster and easier assistance to foreign investigators when a hacker uses a U.S.-located computer as a pass-through to attack computers located outside the U.S.)

The bottom line is that for the purpose of the federal computer crime law, most honeynets are going to qualify as "protected" computers. I next discuss what protected computers are "protected" against.

Denial of Service Attacks and Malicious Code If a protected computer is the victim of a denial of service (DoS) attack or virus, worm, or other malcode with a damaging payload, the attacker may be guilty of a felony violation of the Computer Fraud and Abuse Act. It would not matter whether the perpetrator was an outside attacker who had no right to access the computer, or an inside employee, subscriber, customer, or contractor who had some legitimate right to be on the victim system. Nor is it necessary that the attacker actually gain some level of user privileges to the computer. If an attacker "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," the attacker has committed a felony violation of the law.[22] The maximum penalty for first-time offenders is a fine and 10 years imprisonment. The maximum rises to a fine and 20 years imprisonment for subsequent offenses, or if the intruder knowingly or recklessly caused serious bodily injury, and life imprisonment for knowingly or recklessly causing death.[23] The attacker would also be subject to a civil suit.[24]

20. 18 U.S.C. § 1030(e)(2).
21. 18 U.S.C. § 1030(e)(2)(B).
22. 18 U.S.C. § 1030(a)(5)(A)(i).
23. 18 U.S.C. § 1030(c)(4)(A) & (C).
24. 18 U.S.C. § 1030(g) (allowing for civil suit for compensatory damages, injunctive relief and other equitable relief).

To constitute a crime, the attack has to result in "damage." Under the statute, the term "damage" means "any impairment to the integrity or availability of data, a program, a system or information."[25] By definition, a DoS attack causes damage because it impairs the availability of data, a program, a system, or information. In addition to damage, however, to constitute a felony under this section of the statute, the attack must also have resulted in one or more of the following:[26]

- Loss to one or more persons during any one-year period aggregating at least $5,000
- Modification or impairment, or potential modification or impairment, of the medical examination, diagnosis, treatment, or care of one or more individuals
- Physical injury to any person
- A threat to public health or safety
- Damage affecting a computer system used by or for a government entity in furtherance of the administration of justice, national defense, or national security

Thus, if an attack on a protected computer did not result in impairment of medical records, harm to a person, threat to public safety, or damage to a government entity system, then to constitute a federal crime under the particular provisions I am discussing now, the damage from the attack must have resulted in losses of at least $5,000 in a given year. (Of course, even if the threshold is not met, the conduct may be criminal under some other provision or under applicable state law.)

Any loss that is a reasonably foreseeable result of the attack or incident can count toward the $5,000 threshold. Specifically, the statute defines "loss" to include "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 other information to its condition prior to the offense, and any revenue lost, cost incurred, or other consequential damages incurred because of interruption of service."[27]

25. 18 U.S.C. § 1030(e)(8).
26. 18 U.S.C. § 1030(a)(5)(B).
27. 18 U.S.C. § 1030(e)(11).

In addition, in some situations, an attack on a particular protected computer may not have resulted in much loss when viewed in isolation. To meet the $5,000 threshold, law enforcement can aggregate losses resulting from a related course of conduct that occurs within a one-year period affecting several protected computers and victims. For example, if in a single 12-month period a defendant launches a DoS attack against 11 separate websites, each single website operator may have suffered not much more than $500 loss, far below the $5,000 threshold. In a criminal prosecution of the attacker for the related attacks, however, law enforcement can satisfy the $5,000 threshold by adding those individual losses, the sum of which would exceed the threshold.

Most honeynets are unlikely to have data of any real value, or to offer services to legitimate users, so it may be that most honeynet operators could not show that they suffered significant "loss" as a result of an attack on a honeynet. This does not mean that there was no crime, however. First, as I discuss in a section to follow, there may be a charge for an attempted crime. Second, even if the attack on the honeynet itself was not criminal (although it may be), the attacker may have left valuable evidence on the honeynet that would form the basis for an investigation and prosecution of the perpetrator for criminal attacks on other victim systems.

Intrusions Of course, the Computer Fraud and Abuse Act also covers actual intrusions into a protected computer. If the attacker actually cracks your honeynet and gains user privileges, and as a result causes damage, then the attacker's conduct could also constitute a federal offense (if the intrusion caused one or more of the listed harms).[28] If the damage was caused intentionally, the maximum penalty for first-time offenders is a fine and 10 years imprisonment. The maximum rises to a fine and 20 years imprisonment for subsequent offenses.[29] If the damage was caused recklessly, the maximum penalty for first-time offenders is a fine and 5 years imprisonment. The maximum rises to a fine and 20 years imprisonment for subsequent offenses.[30] If the attacker caused damages neither

28. In sum, those harms are: aggregate loss of at least $5,000 in a given year, impairment of medical records, harm to a person, threat to public health or safety, or damage to a government entity system used in administration of justice, national defense, or national security.
29. 18 U.S.C. § 1030(c)(4)(A) & (C).
30. 18 U.S.C. § 1030(c)(4)(B) & (C).

intentionally nor recklessly, and the attacker is a first-time offender, then the attacker may receive a maximum penalty of a fine and 1 year imprisonment. The maximum rises to a fine and 10 years imprisonment for subsequent offenses.[31]

Other Computer "Access" Crimes Other provisions in the Computer Fraud and Abuse Act prohibit attackers from obtaining information from government systems, financial institutions, and credit card issuers.[32]

There is a violation almost any time a hacker breaks into a computer to obtain information, even if the hacker does not damage the integrity or availability of the data. The crime is more serious if committed for commercial advantage or private financial gain, in furtherance of another crime, or if the information obtained is worth more than $5,000.[33] The maximum penalty for first-time offenders is a fine and 1 year imprisonment (a fine and 5 years imprisonment if committed with commercial or financial motives).[34] The maximum rises to a fine and 10 years imprisonment for subsequent offenses.[35]

It is also a crime under the act to access, without authorization, any nonpublic U.S. government computer, even if no information is obtained nor damage inflicted.[36] Hacking into a computer to further some fraud, and thereby gain anything of value (other than the value of computer cycles themselves), is yet another offense under the act.[37] Computer-related espionage, or obtaining classified information by means of a computer system, is also criminal under the act,[38] and may constitute a federal act of terrorism in certain circumstances.[39]

31. 18 U.S.C. § 1030(c)(2)(A) & (3)(B).
32. 18 U.S.C. § 1030(a)(1) & (2).
33. 18 U.S.C. § 1030(a)(2) & (c)(2)(B)(i)–(ii).
34. 18 U.S.C. § 1030(c)(2)(A) & (B)(i)–(iii).
35. 18 U.S.C. § 1030(c)(2)(C).
36. 18 U.S.C. § 1030(a)(3).
37. 18 U.S.C. § 1030(a)(4).
38. 18 U.S.C. § 1030(a)(1). The maximum penalty for first-time offenders of this provision is a fine and 10 year imprisonment. 18 U.S.C. § 1030(c)(1)(A). The maximum rises to a fine and 20 years for subsequent offenses. 18 U.S.C. § 1030(c)(1)(B).
39. 18 U.S.C. § 2332b(g)(5)(B)(i).

Trafficking in Passwords As a general matter, it is a crime under the statute to traffic in passwords.[40] A honeynet operator should pay particular attention if intruders use the honeynet to store files with names that resemble standard password files. That is a pretty compelling hint that the honeynet is being used to traffic in passwords in violation of the law.

Threatening Damage to a Computer Yet another provision in the Computer Fraud and Abuse Act makes it illegal to extort something of value by threatening to do harm to a protected computer.[41] The statute provides that the communication carrying the threat must have been transmitted in "interstate or foreign commerce," which in practical terms means sent through the mail, by telephone, or in an electronic communication on the Internet. For example, if someone in a chat session threatens to inflict damage to a computer (including a honeynet), intending to obtain money from the system owner, for example, that person may have committed a federal offense. Likewise, if a honeynet operator finds such a threat transmitted to or through the honeynet, the threat may constitute evidence of a crime even if it is a threat to a computer other than the honeynet.

Attempt to Commit a Network Crime The Computer Fraud and Abuse Act also criminalizes *attempts* to engage in conduct that would violate the Act.[42] This means that, although the crime was never completed, a defendant who took a substantial step toward completing the crime but was thwarted may still be charged as if the crime had been completed.

This concept is simple enough for most of the crimes under the Computer Fraud and Abuse Act, even when the victim computer is a honeynet. The crime of attempt becomes a bit more complicated, however, if the conduct the attacker was trying to complete would not be a crime unless it results in damage. For example, to charge a defendant under section 1030(a)(5)(A)(i) for intentionally launching a DoS attack against a company network, the prosecutor would have to show damage to a protected computer. If the attack did not result in damage because it

40. 18 U.S.C. § 1030(a)(6); see also 18 U.S.C. § 1029 (access device fraud).
41. 18 U.S.C. § 1030(a)(7).
42. 18 U.S.C. § 1030(b).

was unsuccessful, the defendant can still be charged with an attempt to launch a damaging DoS attack. Charging a hacker with attempt to commit a crime where damage must be shown may seem a bit odd, however, where the attacked computer is a honeynet. How is it that a honeynet operator (or a prosecutor) can show that damage or loss could have been suffered as the result of a successful attack on a *faux* production server? After all, the typical honeynet will not be populated with data of any real value, and there are unlikely to be legitimate users who are deprived of services as the result of an attack on the honeynet.

Although there is no published case law involving a defendant charged with attempting to attack a protected computer that turns out to be a honeynet, it is not unusual for a prosecutor to charge a defendant with an attempt to commit a crime that, unbeknownst to the defendant, could not have been committed successfully. For example, defendant who attempts to buy illegal drugs from an undercover police officer may be charged with an attempt to traffic in narcotics, even if the police officer did not actually have any illegal narcotics to sell.[43] Likewise, the government can engage in a sting operation to put trade secret thieves in jail without having to put real trade secrets at risk of disclosure.[44] One who shoots a corpse, believing it to be alive, can be charged with attempted murder, and one who sells sugar, believing it to be cocaine, can be charged with attempt to sell illegal drugs.[45]

It may be that even though a hacker is mistaken in believing that a honeynet was a production server chock-full of valuable information, and even if it would have been impossible for the hacker to have actually inflicted any damage or loss on the honeynet or its operators, the hacker may still be guilty of attempting to

43. See, e.g., *Giddings v. State,* 816 S.W.2d 538 (Tex.App., 1991); *U.S. v. Root,* 296 F.3d 1222, 1227 (11th Cir. 2002) (holding "that an actual minor victim is not required for an attempt conviction" under statute prohibiting enticing minor to engage in criminal sexual activity); *U.S. v. Brooklier,* 685 F.2d 1208 (9th Cir. 1982) (impossibility no defense to charge of attempting to extort money from undercover business operated by FBI).
44. *U.S. v. Yang,* 281 F.3d 534 (6th Cir. 2002); *U.S. v. Hsu,* 155 F.3d 189 (3d Cir.1998).
45. See *U.S. v. Lange,* 312 F.3d 263 (7th Cir. 2002). ("Events of this sort underlie the maxim that factual impossibility is no defense to a prosecution for attempt.")

violate the Computer Fraud and Abuse Act by taking a substantial step in committing the crime.[46]

Contraband

A honeynet operator should be ready in the event that the honeynet becomes a repository of contraband. Contraband comes in many forms. Child pornography and other obscene images, stolen trade secrets, pilfered passwords and user names, credit card numbers and account identifiers, and of course pirated software, music, and video are unfortunately common types of contraband that can flow easily over networks.

Crimes Committed by Juveniles

It may be that the crime is committed by a minor. Criminal defendants who are under 18 years of age are treated differently than those who are over 18 at the time of the criminal conduct. Generally speaking, the prosecution of juveniles is left in the first instance to the state courts with jurisdiction over the offense. A charge of juvenile delinquency can be brought against a minor in federal court, however, where the federal prosecutor certifies that: (a) The state(s) with jurisdiction declined to prosecute (or there is no state with jurisdiction), or (b) the state(s) with jurisdiction are not adequately equipped to handle the needs of juveniles, or (c) the crime is a violent felony (or one of the drug or gun offenses listed in the federal statute covering juveniles), or (d) the offense implicates a substantial federal interest warranting federal intervention.[47]

PROTOCOL FOR DEALING WITH ILLEGAL CONDUCT AND CONTRABAND

Before you take a honeynet "live," think about what you are going to do in the event you suspect that your honeynet has become the scene of a crime or con-

46. *U.S. v. Farner,* 251 F.3d 510, 513 (5th Cir. 2001). ("[T]his circuit has properly eschewed the semantical thicket of the impossibility defense in criminal attempt cases and has instead required proof of two elements: first, that the defendant acted with the kind of culpability otherwise required for the commission of the underlying substantive offense, and, second, that the defendant had engaged in conduct which constitutes a substantial step toward commission of the crime. The substantial step must be conduct which strongly corroborates the firmness of defendant's criminal attempt.")
47. Juvenile Justice and Delinquency Prevention Act, 18 U.S.C. § 5031–5042; see *U.S. v. F.S.J.,* 265 F.3d 764 (9th Cir. 2001).

tains evidence of criminal conduct. This is another topic that will be well worth your time to discuss with your lawyer.

Involve Law Enforcement

By its very nature, your honeynet will likely become a victim of or "witness" to criminal conduct. It may be necessary for you to call law enforcement if you see that there is in fact criminal conduct on your honeynet. There are laws that may require reporting certain types of crime.[48] Be prepared in advance of detecting crime.

Establish a Relationship with Law Enforcement If you are a private honeynet operator, one step that is easy and may prove invaluable is to establish a relationship with a law enforcement official who you can call if you detect illegal conduct on your honeynet. There are many avenues available to forge such relationships. The InfraGard program, run out of the Federal Bureau of Investigation, may provide a good entry point to meet federal, state, and local law enforcement in your area who have experience with computer crimes.[49] The field office of the Federal Bureau of Investigation and the U.S. Secret Service closest to you also have agents who work on high-technology crime cases.[50] In some parts of the country, there are Electronic Crimes Task Forces that can provide a great way to meet investigators with the skills to handle cyber crime.[51]

Do not overlook your state and local law enforcement either. Many nonfederal agencies have tremendous expertise in the crimes your honeynet may witness. (Note that if you operate a honeynet in close coordination with law enforcement or other government personnel, there is some chance that a court will conclude that you are an agent of the government and that the Fourth Amendment, discussed above, applies to the honeynet monitoring.)[52]

---

48. See, e.g., 42 U.S.C. § 13032 (those who provide electronic communication services to the public required to report child pornography violations to Cyber Tip Line at the National Center for Missing and Exploited Children); 18 U.S.C. § 4 (whoever, knowing of actual commission of a felony, conceals and fails to report as soon as possible may be imprisoned up to 3 years and fined).
49. *http://www.fbi.gov/contact/fo/fo.htm.*
50. *http://www.fbi.gov/contact/fo/fo.htm* and *http://www.ustreas.gov/usss/field_offices.shtml.*
51. *http://www.ectaskforce.org/Regional_Locations.htm.*
52. See *U.S. v. Jarrett,* 338 F.3d 339 (4th Cir. 2003) (using Trojan horse on defendant's computer, hacker found child pornography and turned over to law enforcement; hacker not deemed "agent of government" because "the Government did not know of, or in any way participate in, the hacker's search of [defendant's] computer at the time of that search").

When to Call Law Enforcement To be sure, not every port scan or worm infection will warrant a call to law enforcement. By the same token, you do not want your honeynet to facilitate criminal activity. If a honeynet operator does not act responsibly when it appears that the honeynet may be aiding a criminal, not only will the honeynet become part of the problem that honeynets generally are intended to solve, but there is a risk that the honeynet operator will be viewed with suspicion. More than a few defendants in child pornography cases have declared, unsuccessfully, that they collected the child pornography found on their computers as part of a "research" project or to ultimately "help" law enforcement. You do not want to play with fire; if you see contraband on your honeynet, do not let the situation go without a response and don't just delete it; get the police involved as soon as you can. Do not wait for the police to call you. As discussed above, by having a solid relationship with law enforcement in advance, the process can be much smoother.

Reduce Risk of Harm to Others

If you find that the honeynet has been compromised and may be used or is being used as an attack platform to victimize other networks, or is being used to distribute pilfered information, you will need to take action to prevent further damage. You certainly do not want to be implicated in the criminal attack on others through your inaction to secure the honeynet. A simple egress filter may do wonders in thwarting an attack. Regardless, consider reporting attempted attacks to law enforcement. Your honeynet may be the only source with records useful to trace the attacker.

Inform Victims

You may discover that the attackers on your honeynet have victimized others, some of whom may have no idea that they have been attacked. For example, you may find that an attacker has stashed on your honeynet a file with credit card numbers and account holder names. Similarly, you may find that your honeynet is being attacked from an upstream source that is very likely itself to be a victim of the hacker. Perhaps, although hopefully not, you may find that your honeynet is being used to attack other networks downstream. In these situations, consider notifying the victims.

Not only will notifying victims allow them to take steps to minimize any loss, they may be able to join the effort to catch the culprit. If you have called in law enforcement, the investigator can often handle victim notification for you. Assistance from law enforcement is particularly valuable in this regard.

Generally, federal government agents have no duty to warn actual or potential victims of activity associated with undercover operations, like honeynets, unless there is some special relationship with the victim.[53] Victim notification may be covered by internal policy, however. Government honeynet operators should not make a judgment call on this alone. They should check with counsel for the agency before deploying honeynets and again if any illegal activity is suspected.

ENTRAPMENT

Entrapment is often mentioned as a concern for honeynet owners. Entrapment is a narrow legal defense that a defendant in a criminal case may raise to escape conviction. It applies where the government acted in a manner that caused an otherwise unwilling defendant to commit the crime charged. If the defendant was predisposed to commit the crime or was not induced by the government to commit the crime, the defense will fail.[54] The government can provide an opportunity and facilities to the defendant to commit a crime; without much more, the defendant will not be heard to claim that he or she was entrapped.[55] The entrapment defense is not based on constitutional rights (unless the operation is so egregious that it "shocks the conscience"). It is really a test to determine whether the defendant had the requisite culpability (state of mind) to be criminally liable for the defendant's actions.[56]

53. *Powers v. Lightner,* 820 F.2d 818, 821–22 (7th Cir. 1987); *Georgia Cas. & Sur. Co.,* 823 F.2d 260, 262 (8th Cir. 1987); *Redmond v. U.S.,* 518 F.2d 811, 816 (7th Cir. 1975).
54. *Sherman v. U.S.,* 356 U.S. 369, 373 (1958).
55. *U.S. v. Hampton,* 425 U.S. 484, 488 (1976).
56. *U.S. v. Poehlman,* 217 F.3d 692 (9th Cir. 2000) (held defendant was entrapped).

The defense of entrapment has no application outside of the criminal process, and in any event, is unlikely to be of much use to a hacker who broke into a honeynet without significant government inducement.

DO NO HARM: LIABILITY TO OTHERS

In addition to the exposure to lawsuits that a honeynet owner faces if he or she violates the statutes discussed above, or contractual rights to privacy, there may be exposure to suits from others harmed by the honeynet. There has been much discussion, for example, about the possibility that a network operator could be sued for having poor security that resulted in an attack against other networks. So far, the discussion has remained largely academic.

Nonetheless, honeynet operators should be vigilant that their honeynets are not used to harm others. A honeynet operator who has configured the honeynet to be vulnerable to an intrusion should pay close attention to activities on the computer. One harmed by such a honeynet may have a field day in court pointing out that the operator intended (and hoped) that the honeynet would be compromised, and in fact made the job of the hacker easier by intentionally including security holes. Yet when the honeynet was exploited as planned, the plaintiff could argue, the operator allowed others to be harmed using the honeynet. (A honeynet run by federal agents, like most undercover operations, will lead to liability for harm done to innocent nontargets only if the court concludes that the government's conduct "shocks the conscience" or is in violation of the victim's constitutional or statutory rights.)[57]

There are several ways to reduce the risk that your honeynet will leave you a defendant in a civil lawsuit.

First, keep a close eye on your honeynet and take action to prevent it from harming others. It is not a fire-and-forget device. The best way to avoid a lawsuit for damage to another's system is to prevent the damage in the first

57. *Brown v. Nationsbank,* 188 F.3d 579, 591 (5th Cir. 1999).

instance. If you have included known vulnerabilities into the honeynet or have otherwise taken steps to drive hostile traffic to the honeynet such that you expect successful intrusions, consider setting up a paging or other notification system so that you will be informed immediately of activity on the honeynet. A good deal of harm to others can occur in just a few seconds, so be prepared to respond without delay. If such vigilance is not practical, consider taking the honeynet offline, or otherwise disabling it during the period that you have no way to attend to it. Make sure that your honeynet is not aiding the nefarious efforts of attackers.

Second, do not just sit on information if you can see that someone is being harmed in spite of your protective efforts. For example, even if you have limited, filtered, or altogether blocked outbound traffic so that the honeynet cannot be used directly as an attack platform, you may find that your honeynet is being used to store hacker tools that are pulled down for use in exploits. Similarly, you may detect an intruder uploading stolen information to the honeynet. Each of these situations holds the potential for a lawsuit. This is when time spent with your lawyer, and contacts with law enforcement can really pay off. Follow the plan you set forth before deploying your honeynet for dealing with criminal activity.

Third, be careful in selecting the data with which you are going to populate your honeynet. Do not populate the honeynet with contraband; it is no more legal for a honeynet operator to do so than the attackers who you hope will find their way to your honeynet.

With careful planning, attention to the legal issues, and close consolation with legal counsel, you can maximize the desired value of your honeynet while reducing both your legal exposure and the risks of harm to others.

SUMMARY

In designing and deploying a honeynet, take the time to consult with an attorney to identify and address the potential legal hazards before they ensnare you. There is no substitute for talking with your own lawyer, who can guide you through the laws that apply to your specific honeynet.

For honeynet deployments in the U.S., consider three legal issues. First, ensure that you are in compliance with the laws that restrict your right to monitor the activities of users on your system. Second, recognize and address the risk that attackers will misuse your honeynet to commit crimes, or store and distribute contraband. Third, consider the possibility that your honeynet will be used to attack other systems, and the potential liability you could face for resulting damage. Your lawyer may identify other legal issues as well. If you deploy a honeynet outside the U.S., look to the applicable laws of the jurisdiction in which you will operate. Designing and implementing your honeynet with attention to these concerns can help you stay out of legal trouble.

The next chapter provides an overview of the types of data that a honeynet may capture and the purpose and value of data analysis.

16 Profiling

Max Kilger, Ofir Arkin, and Jeff Stutzman

As the title of this book suggests, knowing your enemy is a critical component of computer security. The previous chapters have discussed in detail some of the technical strategies, techniques, and issues involved in uncovering unauthorized attempts to penetrate computer networks. In this chapter, the major objective is to convince the reader that identifying and understanding the actors and the motivations behind these activities is just as important as the technical skills, techniques, and tools used to uncover them.

Once an individual or group of individuals has successfully penetrated your network security and compromised a computer, what are their next steps? Those next steps can to a great extent determine the threat level that your computer systems and networks face from a particular attack as well as the subsequent damage that might occur. For example, are these individuals motivated by curiosity and so mainly interested merely in a nondestructive information hunt? The consequences of this kind of intrusion depend greatly upon the nature of the organization being attacked. The extraction of information from a small Florida company making custom lampshades has little impact in the overall scheme of things in the world, whereas even just the briefest exposure to sensitive information stored on a government or military server can have national

security consequences. A honeynet is an excellent platform from which to gather the intelligence necessary to address these questions. It is these kinds of issues and much more that will be the focus of this chapter.

This chapter is organized into three basic sections along with a summary containing some final notes. The first section takes a social science perspective on the social structure and motivators of some types of blackhats so you can gain a broad perspective on their world. The second section describes the life cycle of an attack, detailing the various life stages of an exploit. The third section discusses what might be considered more traditional profiling—the techniques, processes, and logic of extracting and interpreting clues present in the type of exploit tools used, the pattern of their application, and, importantly, the analysis of communications between the attackers themselves. Finally, the last section of this chapter offers two specific examples of profiling and the conclusions drawn from each.

A SOCIOLOGICAL ANALYSIS OF THE WHITEHAT/BLACKHAT COMMUNITY

As noted above, the previous chapters focused on the technical tools and techniques used in helping to detect and identify specific types of attacks and the individuals or groups conducting them. In this section, we will turn our attention to a broader, more theoretical look at the members of this social community. As mentioned in the introduction to this chapter, an understanding of the blackhat community is equally as important as an understanding of the technical tools used to discover their exploits. In gaining an understanding of the blackhat community, we will look briefly at the identity crisis that exists within the community, the motives of individuals and groups, and a look at the social structure of the combined whitehat/blackhat community to identify some of the large-scale forces shaping the attitudes, behaviors, and actions of its members.

The theory and motivations discussed in this section, while specifically aimed at external threats, also generally apply to other specific situations such as insider threats. However, insider threat situations also contain significant intervening forces, such as the nature of the relationship between the company or organization and the employee/attacker. The complex manner in which these intervening

forces interact with the organizational environment lie beyond the scope of this chapter and so are not discussed here.

HACKER, CRACKER, BLACKHAT, WHITEHAT: IDENTITY CRISIS AND THE POWER OF LABELS

At the heart of many of the myths surrounding members of both whitehat and blackhat groups is the extensive history of labeling and mislabeling of groups and individuals that has occurred. Labels are a very powerful component of social life and can have far-reaching consequences for an individual, a group, or an entire culture. They are also a key element in how individuals create and maintain identities for themselves and others. In this case, we are dealing entirely with a latent social label—that is, an identity that is not directly observable. There is no official "hacker identity card," no reliable identifiable physical characteristics (despite attempts by the media to suggest the contrary), nor any single means among members of the community themselves for identifying others that share their identity.

While the latent nature of this social identity makes it easier for individuals to self-identify themselves as hackers, it also presents problems both to the stability of their self-identity as well as to their effectiveness in communicating their identity to others and gaining entrance to a social group of others who also identify themselves as hackers. It also suggests that efforts to produce some sort of objective census of individuals who label themselves as blackhats, whitehats, or some other identity within the hacker community are most likely bound to fail. A brief look at the history of the hacker label will help provide some background on how some of this identity crisis evolved. The origin of the term "hacker" is the computing community itself. The word appeared in early versions of the "Jargon File," a community-maintained file of shared words, phrases, and their meanings which eventually was published in print as *The Hacker's Dictionary,* by Eric S. Reymond (1996). The meaning of the term hacker can be extracted from Reymond's book:

> ":hacker: /n./ [originally, someone who makes furniture with an axe] 1. A person who enjoys exploring the details of programmable systems and how to stretch their capabilities, as opposed to most users, who prefer to learn only the minimum

> necessary. 2. One who programs enthusiastically (even obsessively) or who enjoys programming rather than just theorizing about programming. 3. A person capable of appreciating {hack value}. 4. A person who is good at programming quickly. 5. An expert at a particular program, or one who frequently does work using it or on it; as in 'a UNIX hacker'. (Definitions 1 through 5 are correlated, and people who fit them congregate.) 6. An expert or enthusiast of any kind. One might be an astronomy hacker, for example. 7. One who enjoys the intellectual challenge of creatively overcoming or circumventing limitations. 8. [deprecated] A malicious meddler who tries to discover sensitive information by poking around. Hence 'password hacke', 'network hacker'. The correct term for this sense is {cracker}.

Note that meanings 1 through 7 of the definition do not ascribe intention or assign a moral value judgment to individuals labeled as hackers. In this sense, the term hacker could and originally was applied to everyone who fit the description, whether their actions were viewed as being helpful or criminal.

So how did the term hacker come to have its negative connotations? The news media played a large role in associating criminal behavior with the term hacker. Whenever the news media would report some computer crime-related incident, they would label the perpetrators "computer hackers." According to the formal definition cited above, they were probably perfectly correct in doing so. However, the unfortunate consequence of this repeated news media labeling of criminal incidents as being caused by "computer hackers" was the eventual association by the public of the term hacker with the concept of a computer criminal.

Eventually, individuals within the computer community who called themselves hackers tired of the negative identity they carried and attempted to redefine these "evil-doer" hackers as *crackers,* after their popular pastime of attempting to crack computer-encrypted password files. Further attempts at redefinition followed. More recently, hackers working for goals viewed as positive by society have labeled themselves as "whitehats," while those working for negatively evaluated goals are labeled as "blackhats," in the tradition of the old American West. One natural extension of this nomenclature has been the emergence of "grayhats," individuals or groups whose actions are viewed as somewhere in an ambiguous no-mans-land between the whitehats and the blackhats.

Such efforts made by "hackers" to redefine their identity underscore the importance that labels hold in the world. One veteran member of the hacker community addressing relative newcomers at a recent hacker convention proclaimed "Blackhats, whitehats, we don't wear no stinkin' hats!" to emphasize their rejection of even those labels used within the community itself.

However they define themselves, it is likely that this identity crisis will continue for some time to come. Even today it is common to find references to perpetrators of computer crimes referred to as hackers in the news media, and further instances of this negative stereotyping in the news and popular media are likely to continue. Thus, the stigma that colors the entire computer community is likely to be difficult to shake, even for whitehatted members. In subsequent discussions of motives and social structure, it will be helpful to keep in mind this crisis of identity that many members face.

MOTIVES WITHIN THE COMMUNITY: A KEY TO UNDERSTANDING INDIVIDUALS, GROUPS, AND THEIR ACTIONS

Motivation is one of the most crucial elements in gaining an understanding of why individuals within the computer community do what they do. All but one of the motivations that will be discussed is prevalent in the entire computer hacker community. The exception—money—is most distinctly associated with those individuals who would generally be identified by most people as blackhats. A comprehensive understanding of the six basic motivations within the community will assist computer security professionals in predicting the potential behavior of individuals who gain unauthorized access to their networks. It may also help policymakers in deciding how best to protect the nation's critical information infrastructure given the plethora of threats to many of its key components.

The origins of the six motivations come from an acronym **MICE**—a term long used by the U.S. Federal Bureau of Investigation's counterintelligence unit in outlining the motivations of individuals who commit espionage against their country. The original MICE acronym stands for **M**oney, **I**deology, **C**ompromise, and **E**go. The six motivations we'll discuss in relation to the hacker community are **M**oney, **E**ntertainment, **E**go, **C**ause, **E**ntrance to social group, and **S**tatus—which

forms the allegorically appropriate acronym **MEECES**. A brief discussion of each motivation follows.

Money

Especially in the early history of the hacker community, there was a strong norm against illegally accumulating large sums of money or other financial resources by utilizing one's computer-hacking skills. Note that this norm did not prohibit many individuals in the community from extracting goods or services on a smaller scale. Indeed, many of the best hacks were written to acquire free long-distance telephone service and computer accounts, transportation on metropolitan transportation systems, grade changes inside administrative academic computer systems, and free television programming, among other things. For the most part, these goods and services were acquired for personal consumption by the computer hacker and his or her close friends. Those rare individuals who hacked their way into disproportionately large sums of money, goods, or services in these early days were mostly shunned by the rest of the computer-hacking community.

Today, there is a definite shift in this norm. In recent years we have witnessed numerous incidents in which individuals (who were usually not caught) have tried to blackmail companies (for example, CD Universe) claiming they have hacked the company and have extracted confidential client information from its computing infrastructure. Although some of the extortion cases were hoaxes, in others, the crackers demonstrated (by sending client information back to the companies they had extracted it from) that they had the ability to carry out their threats. In addition, many incidents remain unaccounted for because the hacked companies decided not to report the incident to the authorities or to publicize the incident, but instead to pay "quiet money" to these malicious computer attackers.

Meanwhile, in a spin-off of the extortion theme, captured Internet Relay Chat (IRC) conversations have revealed that rather than extort money from a company, computer crackers are offering to launch denial of service (DoS) attacks against competitors' Web sites as a "business service" to companies.

Credit card theft has also risen dramatically. The ability of blackhats to compromise security features of e-commerce Web sites has enabled the wholesale theft of from hundreds to millions of credit card numbers as well as personal identity

information necessary to successfully complete online charges using these stolen cards. Stolen credit cards have become a "pseudo-currency" or accepted financial instrument within the blackhat community, where freshly stolen credit card numbers may be traded for money, merchandise, accounts on other computer systems, or most any other item of value. There is concern from some quarters that stolen credit cards may become a significant financial instrument funding terrorist groups.

Another alarming phenomenon is the fact that criminal enterprises are hiring talented individuals to perform malicious computer activities to create financial gains for the criminal entities. In some countries, especially in the former Soviet Union, the hired computer blackhats enjoy a rock-star lifestyle as part of their "contract." The rock-star lifestyle includes, in many cases, supply of drugs, sex, Western-made fashions, as well as quantities and types of foodstuffs otherwise unavailable there. The criminal entities in those countries take advantage of the bad economical situation in these former Soviet bloc countries to hire these individuals for criminal enterprises. Incidentally, many of those criminal entities are populated by ex-intelligence community personnel (for example, ex-KGB personnel). There are substantiated reports from expatriate sources from Russia that threats of personal harm to individuals and family members are often an inducement for members of the blackhat community to perform illegal transfers of money or orders for hard goods.[1]

In general, it appears that the proportion of individuals in the hacking community employed in the enterprise of the large-scale theft of money, services, and goods has grown exponentially. Factors including the emergence of the presence of organized crime on the Internet, the global trend of using the Internet to access financial resources and execute financial transactions, and the global exposure of a nation's information infrastructure are all contributing to the increase in popularity of money as a motivation for hacking.

Whereas in the past, individuals whose motivation for computer hacking was monetary were shunned by the rest of the hacking community and often isolated,

1. Based on personal communication to the authors.

they are now present in such numbers that they are able to form their own loose associations and larger-scale criminal enterprises. Indeed, there has emerged a "carder" subculture within the blackhat counterculture where individuals and groups trade techniques, exchange stolen credit card numbers, discuss technical details including defeating various credit card security features, developing automated tools to generate valid credit card numbers, and testing card validity and credit limits.[2]

There appears to be no current or foreseeable inhibitors that may attenuate this trend, so it is expected that money as a source of motivation for hacking will continue to grow unabated.

Entertainment

The motivation of hacking for entertainment is probably the motivation with the least consequences for the intended targets because the final objective of this motivation appears to be more playful than destructive. Early manifestations of this motivation included hapless operators of mainframe computers who watched as large washer-sized disk drives literally walked across the computer room floor coerced by hacker-written code that violently sent the disk drive heads see-sawing back and forth across the disk, or the delight in which a playful programmer programmed a card reader to read a programmer's deck of punch cards and shuffle them out of order into different bins of a mainframe card reader.

Modern-day versions of this motivation still abound. Computer attackers may, for example, hack a company or governmental Web site and post embarrassing pictures or text on the site as entertainment for themselves and their friends. They also may successfully gain access to a mail server and publicly post emails that personally embarrass the company and/or senders and recipients of the supposedly private emails. In other email schemes, they may impersonate the intended target by faking their email address and sending embarrassing emails to everyone in the target's email address book. They may reroute Internet browser

2. For an in-depth look at this newly emerging subculture see the Honeynet Project's white paper "Profiles—Automated Credit Card Fraud" accessible on the Honeynet Project Web site, *http://www.honeynet.org/papers/profiles/cc-fraud.pdf.*

requests for a company Web site to a pornographic Web site instead. They may tap into telecommunications systems and reroute telephone calls for some popular business like a pizza parlor to an unlucky recipient's home phone.

The number of potential schemes deployed in the name of entertainment is pretty much limitless. There is no expectation that this motivation will ever die off, and although its role as a motivator has long been exaggerated in the press as well as in film, it accounts for only a small portion of the motivation behind the actions of members of the hacking community. Still, it is likely to remain a popular one for all colors of the "hatted community."

Ego

The motive of ego is a core motivation shared to some extent by the entire computer hacking community. At the heart of this motivation is the satisfaction that comes from overcoming technical obstacles and creating an innovative solution to a problem. The basic psychological payoff (e.g., intermittent positive reinforcement) of getting the machine to do what you intended it to do is a motivator shared by the whole range of code writers, from the fresh-faced kid writing his very first "Hello, world" program to the "net god" who has just successfully tested a complex multi-threaded, distributed computing process.

Another example is the rise in self-esteem and boost to personal ego that comes from having forced or "tricked" the computer into doing something novel or unauthorized or slipping past complex computer security software without notice. The tougher the technical challenges, the larger the personal payoff when attempts are successful. This motivation is not unique to the computer world; it is often present in many other fields, especially technical ones.

The motivating power of ego is one that should not be underestimated. This motivator often overpowers many other constraints that might otherwise restrain the individual. One of the most common examples of this are the large number of cases where a hacker without any malicious intentions works feverishly and successfully on a method to bypass the computer security on a targeted system such as a governmental or military network. They undertake this objective in the face of the real threat of discovery and apprehension and the subsequent serious legal ramifications.

In light of the power of this motivator, it is unlikely that the new, extremely harsh penalties for computer intrusion outlined in the recently passed USA PATRIOT Act will deter most hackers from breaking into governmental and military systems. Instead, the more likely outcome is that many of these individuals, encouraged and driven by this powerful motivator, will continue to break into these networks, regardless of their intention. Many of them will get caught and be sentenced to harsh prison terms. It seems unlikely, even with the growing trend of longer prison terms that the deterrent effect of harsher penalties is going to have a significant effect, given the power of this motivation over individuals in the hacking community.

Cause (Ideology)

Ideology is often shaped from different factors, such as geopolitical orientation, cultural influences (whether originating from one's own geopolitical location or from social interaction over the Internet), religion, historical events, and views on current social issues. Ideology-driven hacking is a phenomenon that is becoming more common. Often called **hacktivism**, it is the use of the Internet to promote a particular political, scientific, social, or other cause.

Some of the earliest instances of hacktivism come from the hacking community itself. Motivated by the ideology that all information should be free, computer hackers would break into the corporate servers of companies like the various Bell System networks, extract technical information on telephone switching systems, and then publish the information on the Internet for anyone to read and use.

Other instances involved the theft and open distribution of source code for various products, including some early versions of the UNIX operating system, so that others would be free to examine, understand, and even modify the code. In some cases, individuals felt that some commercial software products were so prohibitively priced that they discriminated against lower-income hackers and so they went about writing password cracks and other technical means for disabling copy-protective measures deployed by some software manufacturers. In doing so, anyone who wanted to use the software could, regardless of their financial resources.

More recent instances of hacktivism have involved a much larger scope of causes. One of the most common instances of hacktivism today is the use of

computer technology to denigrate one particular geopolitical position while advocating a competing ideology. Numerous examples of this exist, especially in the Middle East, such as the redirection of the Palestinian Islamic terrorist's group Hamas' Web site to a pornographic Web site, the defacement of the Israeli Likud party leader Ariel Sharon's Web site, as well as mass Web site defacements and DoS attacks between Palestinians, Israelis, and their supporters after Ariel Sharon visited the Temple Mountain. National boundaries are no defense to these attacks, as witnessed by official United States government Web sites like the official White House Web site, which was hacked by Korean computer hackers. The list goes on and on.

A more serious side to hacktivism exists in the efforts of individuals to do more serious damage beyond that of mere political rhetoric. As a recent example, an antigovernment British citizen was indicted for breaking into 92 U.S. government computers—many of them military computers—stealing passwords, monitoring traffic, and destroying files. In another incident during the Balkan War, Yugoslavian hackers not led by the central Yugoslavian government participated in a full-out cyber war against U.S. and NATO cyber targets.

What is beginning to happen is the emergence of "the civilian cyber warrior." In the past, when a citizen of one nation had some objection to ideological, political, or military actions of another nation, there was little that he or she could do without serious financial or physical safety costs, beyond perhaps writing a nasty letter to the head of the offending government or protesting in the streets. Now it is possible for individuals with computer hacking skills to personally strike a blow for their country or cause by looking for vulnerable government and military computers and networks, gaining access and/or control of them, and extracting information sensitive to national security or destroying or altering information in an effort to denigrate that government's military or civilian information infrastructure.

No aspect of a country's critical infrastructure is safe from this form of cyber attack—whether it is a nation's financial systems, the control of their power grid, logistical military information, or the control of transportation systems—all of these systems are to some extent vulnerable to this kind of ideology-motivated cyber attack. All of this can be done right now, and the risk/expense to the civilian

cyber warrior is minimal, especially if their native country is hostile to the United States and refuses to cooperate in potential apprehension or prosecution.

One example of this occurred in 2001 when a US E-3 reconnaissance plane was involved in a midair collision with a Chinese military jet. A group of Chinese hackers, in response to tensions between the United States and China following the collision, began a series of defacement attacks on U.S. Web sites. U.S. computer hackers responded with Web site defacements of Chinese Web sites. This series of cyber attacks escalated with the creation and release of the li0n worm by a member of the Honker hacker group in China. This worm compromised many systems and sent off password and shadow files to a Web site inside China. This may indeed be the first widespread, publicly known real-life coordinated attack by civilian cyber warriors against a nation-state.

The concept of the civilian cyber warrior can now even be extended to those without extensive computer skills. Many of the computer tools used in finding vulnerable computers and compromising them with root access exploits have been "dumbed down" with graphical user interfaces (GUIs) and significant automation so that even individuals with limited technical skills can scan and compromise thousands of computers around the world. This brings up the image of a nation arming its citizens with customized information warfare tools and instructing them to attack a hostile nation's critical infrastructures.

In summary, cause (ideology) as a motivation in the hacking community is one that should be taken seriously. The proportion of incidents and attacks on the Web that are driven by ideology is very likely to increase in the future. In addition, individuals or groups whose behaviors are motivated by ideology are probably likely to resort to more extreme measures, be more persistent, and possibly cause more damage. This is one motivation that bears close watching.

Entrance to a Social Group

People by nature are social animals and therefore have the propensity to gather and form social groups. Individuals in the hacker community are no different in that respect. However, there are social forces operating within the social structure of the hacker community that make joining a group of other like-minded individuals a more involved process.

Part of the complication in joining a group of hackers has to do with the fact that the social structure of the community is a very strong **meritocracy**. A meritocracy is a social structure in which your position within the community depends on the level of skills you possess. Field observations of groups within the community support the idea that hacker groups are very status homogeneous—that is, individuals within a group tend to have technical computer skills that are similar in nature. (We'll discuss the idea of the hacker community as a meritocracy in more detail in the next section.)

In the case of the whitehat/blackhat community, your ability to join a group depends on a match in skills between the existing members and the potential joining member. Individuals whose technical skills are too far below those in the group will be evaluated as "newbies," "losers," or other derogatory terms and will most likely be denied membership. Those individuals who have technical skills much higher than those within the group are not likely going to want to join that group.

One problem that often complicates the joining process is that there is no direct measure or indicator of the technical skills of the potential member other than examples of code or other exploits that the joining member has authored that provide evidence of his or her technical expertise. Thus, one of the motivations of these potential members is to write elegant code that demonstrates their technical expertise. This motivation applies to entrance to both whitehat and blackhat groups. Thus, writing a particular exploit, defeating a particularly strong computer security defense, writing some stealthy piece of code that surreptitiously monitors network traffic—all these provide evidence of technical skill levels that can be evaluated by other members of the group.

Status

Status is by far the most powerful social force within the social structure of the whitehat/blackhat community. It motivates more of the behavior within the community than perhaps any other component. As discussed in the previous paragraphs, the hacker community can be characterized as a strong meritocracy. Your position within the status hierarchy of the community depends on your technical skills in coding, network protocols, and other areas of technical expertise.

One of the difficulties that individuals face in a strong meritocracy is communicating status position in a social group that often exists almost exclusively in cyberspace. Much of the real-time communication between members of a hacker group comes in the form of IRC chats, where members of a group can gather in cyberspace to discuss and communicate. One of the problems here is that many of the information clues or "status markers" that are normally exchanged in face-to-face interaction are verbal or nonverbal in nature. Behaviors such as speech rate, eye gaze while speaking or listening, and many other clues are absent in IRC chats. Therefore, members have to resort to other means to broadcast their status position within the group to others. This may take the form of bragging about how many systems they "own"—which, of course, fuels the motivation to compromise computer systems in order to make a valid claim.

Other status markers in IRC chats include the disclosure of knowledge to another group member. Often on IRC chats you will observe one member teach another member how to gain root access using a specific exploit or vulnerability. They also share with others information on exactly how some operating system internal works in order to figure out how to compromise it.

Given the ambiguity inherent in typical communications among members of a hacker group, it often comes to pass that there are status conflicts in which two individuals each believe that they are the more skilled and thus the higher-status individual. Often this status conflict leads to some harmless exchange of derogatory remarks and comments about the skills or expertise of the other individual. Sometimes this conflict may escalate into a contest where one of the individuals will attempt to trash the personal machine of the other member or capture the machines they already have compromised and thus force the other member to "lose face" and, consequently, status. The status conflict may also erupt into a contest that spills out onto the net, where they may compete to own the most machines or fight to see who can bypass the security of a chosen site first.

These status conflicts not only occur within groups, but often between groups as well. Often there is a sense of competition among hacker groups in terms of who is the most skilled. These inter-group conflicts often evolve into miniature "wars" where there is a concerted effort on the part of one hacker group to disrespect the

other group, take over machines belonging to the other group, or otherwise do damage to their status position.

It is interesting to note that in respect to status conflicts between whitehat/blackhat groups, the hacker conventions that convene each year provide a critical function in reducing this conflict. These conventions give the groups the opportunity to meet face-to-face, exchanging not only verifiable information about the skills and expertise of members of their group, but also exchanging those critical verbal and nonverbal status cues that help establish the status hierarchy among them.

Finally, it should be noted that status processes active in the insider threat environment operate differently than in the external threat situations just described. Often in the case of a malicious insider, the comparison actor in terms of evaluating relative status is the company or agency the insider works for. Insiders may compare their personal level of expertise in programming or information networks with some imagined level of general corporate competence and find that they evaluate themselves as more competent than the company or agency that employs them.

This higher level of personal competence that insiders feel in relation to their organization leads to their evaluation as having higher status than this corporate social entity. When the company behaves in a manner that is status inconsistent—such as failing to listen to their advice—this produces status conflict that in turn leads to affect processes that produce feelings such as anger. This may in turn trigger social control processes—similar to ones present in previous discussions of individual blackhats—where the insider attacks the organization's information network as a means of demonstrating and reestablishing his or her position as a higher-status social actor.

SECTION SUMMARY

In summary, in looking at the actions and behavior of individuals within the hacker community, it is important to frame them in terms of the six motivations presented in this section. These are, of course, not all the motivations present within the community, but they are what we consider the major ones present. In the next section, we will take an even broader look at the hacker community and

its social structure to understand more about this fascinating group as well as examine the difficulties in studying this often hard-to-reach group of individuals.

THE SOCIAL STRUCTURE OF THE WHITEHAT/BLACKHAT COMMUNITY

Studying the Community Is Not as Easy as It Seems

The inimitability of the social structure of the computer hacker community and the complex nature of its members makes it one of the most fascinating cultures imaginable. What to the untrained observer looks to be a chaotic community distinguished by a distinct lack of rules, organization, or common objectives turns out instead to be a culture where there is in fact a robust social structure containing strong norms, tight-knit social groups, and a persistent sense of solidarity, when examined carefully.

The key phrase here is "examined carefully." Even a well-trained social scientist will encounter serious difficulties in researching the hacker community. You can't just walk up to members of a blackhat group and announce "Hey, I'm a social scientist. Can I come hang out with you guys?" Well, you could, but you would probably get the same treatment one of the authors of this chapter saw a hapless reporter receive at one hacker convention when he sat down at a table full of pretty serious hackers and posed the question "Hey, hack any good machines lately?" Everyone at the table glared at him and proceeded to studiously ignore him for the remainder of the lunch.

So how does one study this fascinating community? There are a number of methods that work with varying levels of success. One of the authors of this chapter saw social scientists attempting to administer a survey to hackers at a national hacker convention. At best a convenience sample, this type of quantitative research methodology really can't be counted upon to make statistical inferences about the hacker community in general, and probably would not even produce a very valid picture of hackers attending that particular convention.

Qualitative research methodologies are much more suited to investigating the social structure and members of this community. For example, field observation where the researcher is immersed in a native environment such as attending a

hacker convention is a very productive method for gathering data to better understand the social interaction among members. Even this method has its limitations. The community is understandably suspicious of out-group or nonmembers given the pressure exerted on its members by various law enforcement and intelligence agencies. The widely accepted statistic at the annual DEFCON hacker conventions in Las Vegas is that one out of five attendees is a "Fed," an employee of a federal agency. Thus, attempts by a field researcher to videotape or audio record activities, events, groups, or individual members associated with the hacker community are usually met with failure and expulsion from the venue. Even taking notes in a field notebook is often looked upon with suspicion in such an environment where it is not unheard of to observe more "endangered" individuals attending wearing ski masks, paper bags over their heads, or Halloween masks to protect their identities.

Interviewing members of the hacker community is also a fruitful method for gathering information that can help the researcher better understand its members. Once again, there is a great reluctance on the part of members to talk with someone unknown to them. Often it is necessary to use connections within the community who can vouch for the researcher in order to gain the trust and eventual interview. Discussions with members can be very informative and provide very valuable insight into the life of members of the community. Here again, it is best for researchers to rely upon their memory and leave the notepad behind for post-interview reconstruction of the interview. Gaining the trust and respect of members of a particular blackhat group can open up new perspectives on many aspects of their world and significantly help researchers better understand the culture.

Finally, there is a lot of documentary evidence available on the Web for researchers to examine. Web sites, IRC chats and chat transcripts, list discussions, and even videos from hacker conventions can serve as avenues to gather information to make sense of the community. This kind of documentary evidence should not be ignored as it can provide a unique look at the community without the obtrusiveness of interviews or field observation.

A Look at the Hacker Community

One of the first things to note is that the hacker community is not truly a subculture but rather a counterculture by definition. A **subculture** is a culture where members have nonconflicting norms and values that distinguish them

from others but do not conflict with other mainstream societal norms and values. A counterculture is a culture where the norms and values of the culture stand in opposition to traditional society. The definition of counterculture seems a much better fit for the hacker community.

Inside this counterculture is a community filled with individuals with some very unique traits. These traits include the propensity to focus on the details and technical aspects of projects and activities without putting their activities into a larger social perspective. The serious intensity with which a blackhat works on a particular exploit often overshadows an objective perspective of the multidimensional (economic, political, social, and military) collateral damage the code may eventually do, often at the hands of individuals other than themselves.

They also show a resilient willingness to suspend belief in and awareness of significant adversarial social institutions even in the face of serious consequences. For example, they aggressively participate in activities that are clearly illegal and are associated with serious punishments, yet they often ultimately fail to accurately evaluate the probabilities of getting caught and almost assuredly receiving significant punishment by a very unforgiving legal system. In this sense, they share some of the same denial of consequences seen in other youth-oriented legal and nonlegal group enterprises.

Finally, like other countercultures, the community is filled with individuals who share an uncommon, single-minded passion for the values and norms of their counterculture, a passion not often seen in traditional, mainstream society. It is this sense of intense curiosity, fascination with, and feeling of belonging in the digital world that seems to bond them together as a community, even under significant pressures from a number of considerably powerful social institutions. These individuals seem to have found a level of Durkheim's sense of mechanical solidarity usually reserved for preindustrial revolutionary societies encased in a modern world characterized by the organic solidarity of specialization and the division of labor.

The Community as a Strong Meritocracy

As noted above, a common misconception about the hacker community is that it is a community filled with chaos, lack of structure, an abundant lack of rules, and

riddled with "antisocial" individuals. The fact of the matter is that within this community there exists a very strong social structure complete with strict norms, widely shared values, complex organizational structures, and populated with individuals who seek out other members in search of social solidarity.

As we mentioned above, the social structure of the hacker community can best be described as a strong meritocracy—that is, a society where the skills and expertise of an individual have wide-ranging consequences on their status position within small social groups as well as the larger community as a whole. Meritocracies in and of themselves are not unusual entities—mainstream society in the United States can be described as a milder form of meritocracy, where status and rewards are allocated based at least in part on skills and expertise possessed by its members. However, the variant of meritocracy in the hacker community is characterized by a much stronger effect of technical skill and expertise on an individual's status than even in a postmodern industrialized country like the United States.

Given the emphasis on status within this social structure, individuals within the hacker community spend a significant portion of social resources attempting to communicate their level of status to others both within their local social group as well as outside of it. This may take the form of identifying themselves as a member of a particular hacker group. There may also be self-sourced efforts aimed at communicating levels of status such identifying themselves to others as elite (often communicated as el33t, l33t, le3t or some other combination of symbols).

Communicating one's status may also take the form of boasting about the personal possession of large amounts of one type of blackhat currency or another such as the number of stolen credit card numbers, stolen ISP accounts, machines they "own" (have current successful exploits installed on) or possession of a particularly rare or powerful exploit or hacking tool.

Another more effective means of establishing one's status in the community is to author what is judged by others to be an elegant hacking tool. The level of status that can be attained in this manner depends upon the difficulties overcome in making the hacking tool work, the stealth of the tool, its ability to circumvent security measures—all of these factors impact the hacking value and subsequent status value of the tool that is imparted to its author.

Status processes are in play especially within social groups in the community. There are a large number of loosely organized whitehat, blackhat, grayhat, and other colored-hat social groups present within the hacker community. These social groups are different from, say, special interest groups in mainstream society as the groups tend to meet virtually through IRC chat rooms rather than face-to-face, as do most other traditional special interest groups. The hacker groups themselves tend to be status homogeneous to a great extent, with membership composed mostly of individuals with similar levels of technical expertise or skill. Entrance to membership in the group is often by invitation and often only extended to those individuals who possess technical skills at a level similar to others in the group. Individuals with lower levels of skill, and thus in the meritocracy possessing lower status, sometimes attempt to join higher status groups. These upper mobility seeking individuals are often rebuffed with labels of "newbie," "loser," "lame," or other derogatory terms as previously stated.

There is a significant amount of derogatory behavior associated with status communication and conflict within social groups in the community. The combination of constrained communication channels (IRC, email, phone bridges, or even video streaming with inexpensive "golfball" digital video cameras) that fail to convey significant pieces of verbal and nonverbal status information combined with the inherent close status positions of individuals within homogenous hacker groups greatly enhances the occurrences of this type of derogatory behavior. These status conflicts are most often resolved through means such as physically meeting in a place. This meeting may take the form of something as informal as a *2600* monthly meeting at a public mall or corporate building lobby or as organized as traveling from all over the country to meet at a national hackers convention such as DEFCON, the HOPE series on the East Coast, or several other meetings that will remain anonymous. Regardless, status conflicts within groups remain a serious disruptive force that often threatens to seriously degrade the social cohesion of the group to the point where the group disintegrates in a cloud of in-group factional fighting.

External Forces that Affect the Community

There are many external factors, such as geopolitical forces, that have an impact on individual members and groups within the hacker community. The very active Romanian blackhat community is one example of how the geopolitical cli-

mate can affect motivations and behaviors within hacker community. Prior to 1989, Romania was a development center for computer technology and software to Eastern Bloc countries. This technology infrastructure was supported by strong Romanian university programs in mathematics and the sciences. Current political and economic conditions within Romania are quite bleak. There was rampant large-scale economic inflation of approximately 34 percent in 2001, and, according to the CIA Factbook, 40 percent of the people lived below the poverty line in the year 2000. There is significant unemployment among highly educated and technically trained individuals and widespread corruption among various government agencies. While only about 3.5 percent of the population has Internet access, Romania remains a hotbed of blackhat hacker activities.

Ego-motivated hacker attacks are encouraged by the lack of legitimate outlets and rewards for technical skills that lead to high levels of frustration and need to "prove technical expertise" and restore self-esteem. Secondly, the motivation of cause/ideology is present in that a sense of global relative injustice may encourage these individuals to attack targets in countries where their technical skills are more valued and rewarded. Most importantly, money is a major motivator for Romanian blackhat groups. The extremely distressed Romanian economy is encouraging large numbers of these highly skilled individuals to apply their technical computer expertise towards illegal hacking that results in the capture of financially viable resources such as credit card numbers, rerouted materials, intercepted or reconstructed wire transfers of funds to and from financial institutions, and so on.

Mapping the Social Structure of the Community

So far we have suggested that the social structure of the hacker community is a complex one with a number of important elements that influence the motivations, thoughts, and actions of its members. The question becomes, how can one create an overall picture of this counterculture that details its important components?

An initial answer to this question is provided by a tradition that most cultures and countercultures participate in: the creation and maintenance of a cultural historical record. Traditional societies accomplish this task through historical books, museums, and educational institutions. In the case of the hacker community, a written record of historical events as well as concepts, ideas, and people important to the counterculture was created many years ago and has been maintained to this

date. The record we refer to is the "Jargon File," which, as we defined above, is a computer file consisting of a compilation of the important details of the hacker counterculture created and edited by members of the community. Present online on many mirrored servers, as noted above, the Jargon File eventually ended up being published in print as the *Hackers Dictionary.* This document is of significant importance as an anthropological and sociological source of information about the social structure of the hacker community.

We undertook an analysis of the Jargon File to see whether we could identify important dimensions of the social structure of the hacker community. We conducted a two-phase content analysis of the distinct words or phrases contained within the Jargon File. The first phase utilized open coding and uncovered 20 thematic categories within the Jargon File. The second phase utilized axial coding and combined several themes and identified 18 distinct thematic categories within the Jargon File. We had identified 9 of the 18 thematic categories through previous field observations of the hacker community. The remaining nine thematic categories emerged *sui generis* from the analysis.

A total of 1,989 entries consisting of words or phrases were available for analysis in the Jargon File. We determined that approximately 17.8 percent of the words and phrases did not belong to any of the existing 18 categories, could not be combined with other unclassified items into a new category, or were close synonyms for other words or phrases in the Jargon File and were determined to lack sufficient description to classify them as a distinct word or phrase. We classified the remaining 1,635 items into at least one of the 18 different categories. Note that it was possible to code a word or phrase as belonging to more than one thematic category. The nature of the categories themselves contribute to the basic understanding of the social structure of the hacker counterculture. The following are the thematic categories and brief descriptions of each one:

- **Technical.** Having to do directly with some technical aspect of computer hardware, software, algorithm, or process. Example: *kamikaze packet,* a network packet where every option is set.
- **Derogatory.** A word or phrase that is used in a derogatory fashion towards a person or object. Example: *bagbiter,* software, hardware, or a programmer that has failed to perform up to standards.

- **History.** A word or phrase that refers to a specific event, person, or object in the past that is deemed to be of sufficient significance that the typical hacker would have some generalized knowledge about it. Example: *The Great Renaming,* the day in 1985 when a large number of newsgroups on USENET had their names changed for technical reasons.
- **Status.** A word or phrase used to note the status of or esteem with which a person, event, or object is viewed by others in the hacker community. Example: *net.god,* a person who has been using computer networks (USENET, etc.) for quite some time or personally knows one or more individuals of high status within the hacker and computer community. The term also traditionally implies expert technical skills.
- **Magic/Religion.** A word or phrase that explicitly refers to magic or some individual, object, or event with paranormal powers or characteristics. It can also be a word or phrase implicitly or explicitly describing events that cannot normally be explained. Example: *incantation,* some obscure command or procedure that does not make sense but corrects some software or hardware problem.
- **Self-Reference.** There are two instances where this category applies. In the first instance, the word or phrase refers to a characteristic of a computer that a person ascribes to themselves or another person. The second instance refers to the anthropomorphic practice of assigning human traits to computers. Example: *pop,* which refers both to an operation that removes the top of the stack of a computer register or to someone in a discussion suggesting that the level of detail of the conversation is too deep and should return to a more general level.
- **Popular Reference.** The use of popular culture concepts or characters in describing something in the social world of the computer hacker. Example: *Dr. Mbogo,* a professional person whom you would not want to consult about a problem. Taken from the original Addams Family television show, Dr. Mbogo was the family's physician who was portrayed as a witch doctor.
- **Social Control.** Words or phrases that are directly used in a social control process. Example: *flame,* an email message that holds its recipient up to ridicule.
- **Humor.** Words or phrases that are direct attempts at humor are put into this thematic category. Example: *Helen Keller mode,* a computer that is not responding to input and not producing any output.

- **Aesthetic.** An object, event, or process that is thought to have elegant qualities. Example: *indent style,* the practice of using a set of rules to make a computer program more readable.
- **Communication.** The use of computer terms in actual speech between two or more individuals. Example: *ACK*, a data communications term meaning that one computer acknowledges the communication of another computer. Also used by individuals in the hacker community in conversation to acknowledge a statement made by another.
- **Symbol.** Any symbol that has meaning beyond its strict technical interpretation. Example: *bang,* the exclamation point symbol (!) that is used in email addresses and in computer languages.
- **Measure.** Any word or phrase that denotes a certain level or unit of measure. Example: *byte,* a unit of memory consisting of 8 bits.
- **Social Function.** The deliberate use of a word or phrase by a hacker to describe some aspect of social interaction. Example: *lurker,* an individual who reads a newsgroup regularly but rarely or never contributes to it.
- **Metasyntatic Variable.** A letter or word that stands for some variable quantity or characteristic. Example: "If we had done *x,* nothing bad would have happened," referring to the idea that if they had performed some specific yet unnamed action, then the unwanted event would not have happened.
- **Recreation.** Words or phrases that refer to play or leisure activities. Example: *Hunt the wumpas,* a very early computer game played by hackers.
- **Book Reference.** A word or phrase that refers to some specific book. Example: *Orange Book,* a U.S. government publication detailing computer security standards.
- **Art.** Words or phrases that directly refer to some artistic element or object. Example: *twirling baton,* an animated graphic often found in early emails.

Even a quick analysis of the thematic categories that comprise this counterculture suggests a complex social structure with a rich retinue of social elements that play a part in the community. Even more enlightening is the distribution of the instances of these thematic categories. Figure 16-1 illustrates this distribution and lays open components of the social structure of the community somewhat like the rings of an onion.

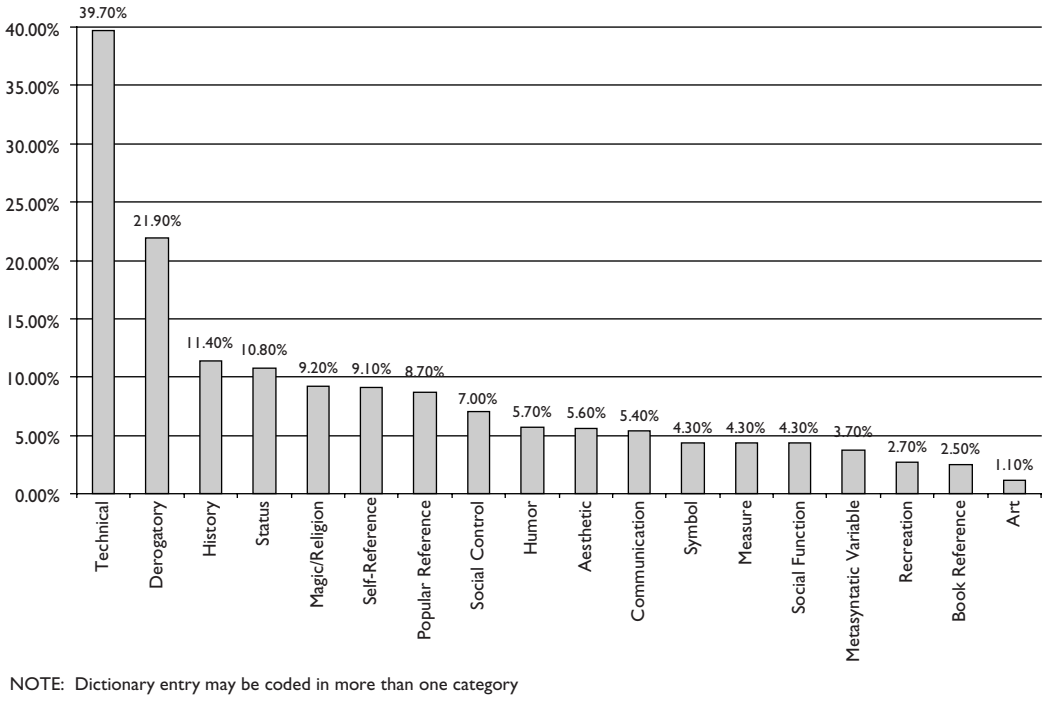

Figure 16-1 A thematic analysis of the Jargon File

As you might expect, the largest thematic category is the technical category. Much of the communication among members of the hacker community deals with technical details of operating systems, networks, programming languages, and so on. The next largest category is the derogatory category. This result is in line with our previous discussion about the abundance of status conflicts and the derogatory exchanges and actions that result from those conflicts.

History is the next most popular theme, which suggests that as a newly emerging counterculture, the hacking community has a need to create a historical record of the birth and growth of the counterculture.

As suggested by our earlier discussion about the merit-based nature of the hacker community, status is the fourth largest thematic component in the analysis. Status

as a social force within the social structure is a very critical component of this counterculture and reinforces the idea that much of what occurs within social groups and between individual members has a significant status component.

One of the more surprising (and prominent) thematic categories to arise from the analysis is the magic/religion category. While this was one of the *a priori* thematic categories that we anticipated would emerge from the analysis, it is one that often surprises people who are not familiar with the hacker community. The most common comment that arises when this result is discussed is "You mean hackers are religious??? You've got to be kidding."

The answer to this quandary can be found in the nature of the technology that lies at the heart of this counterculture. Many members of the hacker community deal with complex operating systems, program applications, and network architectures where it is often not possible to answer with certainty the question "If I perform action A, will the operating system/program/network behave precisely with result B?" That is, because of the complexity of modern operating systems, programs, and network topologies, there is a disconnect between the classical forces of cause and effect. Whenever you have a situation where you cannot logically reconstruct the linkage between cause and effect, you in effect have an instance of "magic."

The instances of magic and magical powers abound in the hacking counterculture. The designation of an individual as a network "wizard" or the often used concept of chanting (e.g., issuing some command to an operating system in hopes of getting it to work correctly) are some common examples. Even official manuals for operating systems put out by large Silicon Valley corporations such as Sun Microsystems refer to elements like "magic numbers" when configuring file systems. One long-running hacker group had (and perhaps still has) an individual filling the position of "high priestess" for the organization.

SECTION SUMMARY

This discussion has just begun to scratch the surface of the social structure of the whitehat/blackhat community. As the counterculture grows and changes, there are undoubtedly going to be many more surprises and opportunities to better

understand this unique community. We hope that this discussion has dispelled some of the stereotypes that dog members of the community as well as provided some analytical insight into the components and forces that shape the community each day.

"A Bug's Life": The Birth, Life, and Death of an Exploit

Where do the exploits that blackhats use come from? While there may be some precedent for answering "from the cabbage patch"—the classic answer to any perplexing origin question—that answer certainly isn't going to help us here. In this section, we take a look at how exploits emerge, metamorphose, spread, and finally expire, if in fact they do indeed expire.

The first two terms that you need to get comfortable with are vulnerability and exploit. A **vulnerability** is a security flaw found with a certain technology. The technology can be an operating system, an application program, a network protocol, a mathematical algorithm, or sometimes a hardware component, such as the buffer memory of a network interface card. An **exploit** is a computer program that is designed to take advantage of a vulnerability with the goal of giving the user of the exploit any one of many potential objectives. These objectives range in seriousness from critical root access to a computer (which gives the attacker complete control over the compromised machine) to some simpler, less-threatening objective, such as access to files on a computer system. The actual objectives and the subsequent damage that is done depends on many factors, including the motivations of the attacker and the value of the information present on the compromised machine.

The Discovery Stage: Finding a Vulnerability

One of the first steps in the birth of an exploit—finding a vulnerability—usually is not a trivial task. The process of uncovering a previously unknown vulnerability usually involves some serious research into the internals of an operating system, application, or other potential vulnerability vector followed by some period of code writing. During code writing, the core or "engine" of the potential exploit

is written and tested against the vulnerability to see whether the vulnerability is real and whether an effective exploit is a logistically reachable goal. It may be the case that the vulnerability is easy to find but difficult to take advantage of, or it may be the case that the vulnerability is difficult to find but once found taking advantage of it is relatively easy.

Also involved at this stage is the skill level of the individual or group pursuing this path. Higher levels of skill on the part of the individual or group result in the discovery of more deeply hidden vulnerabilities. Those with high levels of skill also obviously have a better chance of having the expertise necessary to take advantage of these vulnerabilities.

TECHNIQUES IN FINDING VULNERABILITIES

There are a number of techniques used in finding vulnerabilities, ranging from the simple to the complex. **Source code auditing** is a technique wherein the individual blackhat or group has access to the source code for an operating system or one of its components or an application—often the case where Open Source software is being developed. A careful analysis of the source code can reveal areas of the code that are vulnerable to attack.

When source code is not available, another technique that is used is reverse engineering, which we discussed in Chapter 14. In **reverse engineering**, a special tool is applied to the binary code from the application or operating system component, and often a pseudo-representation of the application code is reproduced for locating weak spots in the code.

Analyzing network traffic can also sometimes be useful in finding vulnerabilities. By capturing and analyzing the network traffic exchanged between computers on a network, it is possible to identify flaws in the communication mechanisms. Researchers might perform different tests to reveal security-related issues with the targeted system, including modifying captured information that will be later resented. Generating carefully crafted traffic against the targeted system may lead to unexpected behavior from the targeted system, and might reveal problems with handling different types of packets or values inside different fields of a

packet. One might also analyze the captured data for known issues with the communication protocol used such as the use of clear text protocols for sending authentication information.

In other cases, attackers may resort to a "black box" technique where nothing is assumed to be known about the computer under scrutiny. In this technique, a close analysis of the inputs and the responses from the targeted computer are used to gather information about operating system type and version, hardware platform, and other items in order to identify the system so that vulnerabilities known for that class of machine can be examined. A top-down analysis is a similar technique in which the characteristics of the machine (operating system, hardware platform, and so on) are already known and the attacker starts from there to examine already-known vulnerabilities to be tested on a particular target machine.

The simplest technique for identifying vulnerabilities is simply to search the Internet for bug tracking lists ("bugtraq lists"), software company public announcements, and even internal corporate communications that document vulnerabilities. With a little social engineering and a simple phone call to a technical support line, a blackhat might unearth a critical vulnerability in a system.

THE PROCESS OF FINDING A VULNERABILITY

The act of finding a vulnerability can be a solitary one or a group effort. It is not that uncommon for blackhat or whitehat groups to hold "hackathons" where they will spend 12 to 72 continuous hours working a specific operating system or component looking for a vulnerability. The nature of the actor looking for vulnerabilities is variable as well. A vulnerability hunter might be a software company auditing a new product it is about to release to the market. They might also be an end-user organization wishing to test the security of some application they are considering rolling out into their enterprise. More malevolently, the vulnerability hunter might be a blackhat or blackhat group looking for vulnerabilities that might be exploitable. Whether a solitary or group effort, the general process of finding a vulnerability is summed up in Figure 16-2.

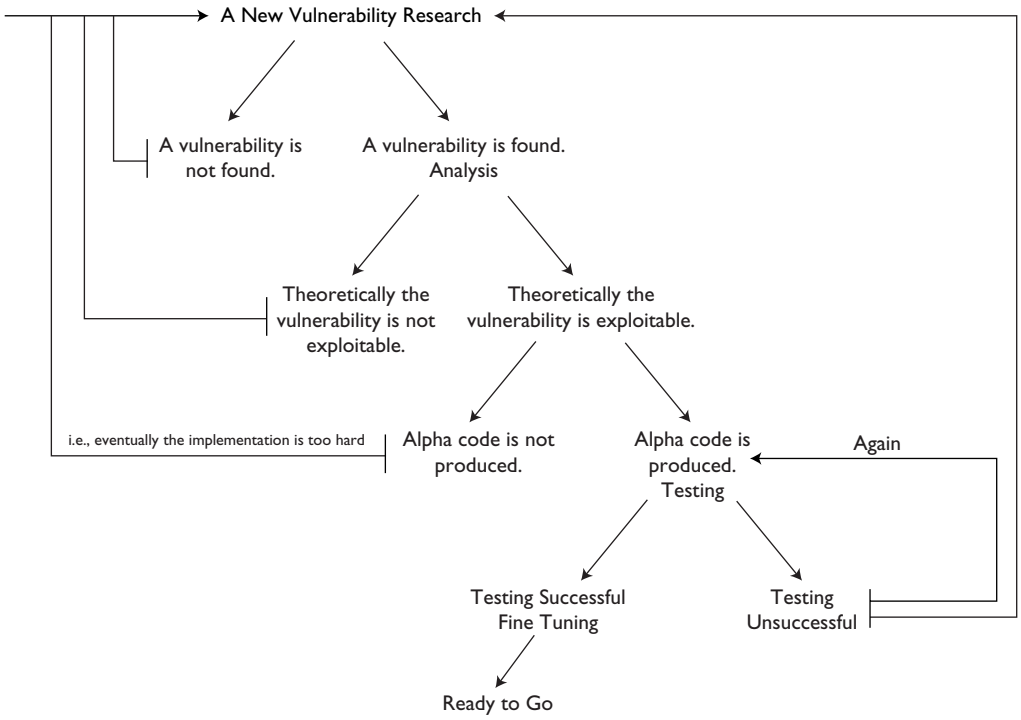

Figure 16-2 Finding a vulnerability

THE BIRTH OF THE EXPLOIT

During the research for a vulnerability and/or the development of an exploit, the knowledge of the vulnerability or exploit's existence is either the sole property of one individual or it might be shared among a small group of individuals that has some kind of relationship and/or interaction with the vulnerability researcher/initial exploit writer (i.e., they are members in the same hacking group).

The exploit is often held tightly by the sole author or small hacking group because of the potential value the exploit has. A newly discovered vulnerability has a high value in the "commodity market of the security world." A zero-day exploit (defined as computer language code written to take advantage of a particular vulnerability that has been discovered but is not publicly known) can be

traded for other exploits and/or other goods (i.e., money). The exploit developer might offer his or her merchandise for sale. The buyers might come from a variety of entities—criminal enterprises, commercial companies,[3] and even government agencies. The potential buyers/traders may use the exploit in a number of ways: diverting monetary funds, gathering information about competitors, or even gathering intelligence information from foreign nation-states.

In addition, the probability that the original author of an exploit will be identified and subsequently apprehended by law enforcement officials is often near nil if the author is reasonably careful about his or her activities. There are several reasons for this lack of successful prosecution. The first is that unless the exploit is discovered by industry security professionals or law enforcement agents early in the life cycle of the exploit, the exploit will have been distributed so widely that tracing its origins would be nearly impossible. The second reason deals with the lack of federal law enforcement agents suitably skilled and trained to investigate this type of crime.

The exceptions to this are situations where very widespread economic damage is sustained due to the new exploit, the exploit has been implicated in the breach of some governmental or military computer systems that could compromise national security, or the exploit author has been careless in his or her actions or communications. These exceptions often attract sufficient attention from the computer security industry and law enforcement that they result in the rare apprehension of an exploit author. The message here is that exploit authors often walk away scot-free.

The original exploit code or vulnerability may be shared by the original author with members of his or her social hacking group. This sharing may occur because exploiting the vulnerability may be beyond the original discoverer's skills. It may also occur because the original author has created the kernel of the exploit and needs assistance in either fixing parts of the exploit that are not working correctly or developing the exploit further. It may also be shared to have it tested against or ported to a hardware or operating system platform that the original author does not have access to. It is also likely, even if the original author

3. Evidence gathered from the Underground community.

develops a fairly complete exploit, that it will eventually be shared within his or her social group as a pathway to enhance his or her status within the group by demonstrating expertise in the form of authoring the exploit.

THE INITIAL DEPLOYMENT OF AN EXPLOIT

The exploit is likely to be closely held by the original author or small, isolated hacking group only for a limited period of time. The factors involved in the spread and discovery of the exploit are numerous. These factors include:

- The service or systems the exploit targets
- The damage the exploit will cause
- The outcome of using the exploit
- The type of the exploit (remote exploit versus local exploit)
- The quality of the exploit
- The ways to discover the exploit
- The ways the break-in(s) will be hidden
- The individuals behind the development of the exploit (individuals in the underground versus a commercial entity)
- How many people will control the exploit
- How many people will be knowledgeable about the vulnerability
- How the exploit will be used (massively versus selectively)

In addition to these factors, there are also psychological factors involved in how the exploit is deployed and eventually escapes the control of the original author or hacking group. The status value of the exploit makes it tempting for the author or individuals in the group to disclose it to others in attempts to raise their status either within the original hacking group or among a larger segment of the hacking community.

EXPLOIT DISCOVERY

In most cases, the exploit code or knowledge eventually is leaked by one of the group's members or is discovered by the security community (i.e., it is left on a

compromised machine used in a penetration test against a prime target that will share the information about the newly developed exploit with other commercial entities). In most cases, the security community will discover the "problem." How long it takes for the security community to find the vulnerability is a different question. Factors that contribute to the discovery of a new exploit by the security community include (but are not limited to) an increase in traffic for a particular service or port, error messages in log files for particular services, and a realization of a correlation between recent unresolved incidents noted by members of the computer security community.

In some cases, exploit code or vulnerability information that has been developed in a commercial environment leaks to the underground and is used maliciously. However, usually the opposite happens—exploit code and technology developed in the hacking community is leaked or revealed (through friendships or through commercial relationships) to commercial companies. There are more and more cases in which a member of a hacker group has used a vulnerability that was found by another member of the group and shared among all in order to secure employment in a computer security consultancy firm. The computer security consultancy firm uses the information in a public relations effort and claims the vulnerability was found by one of the members of its "research team."

Parameters that Contribute to Discovery

There are a number of parameters that usually contribute to the discovery of the fact that a new exploit is being used in the wild. One of these parameters is the ability to capture the network signature of an exploit. If one of the networks the exploit is being used against has a proper monitoring system, this might be a possibility. There are also organizations such as CERT\CC (Carnegie Mellon's Coordination Center) and SANS (Sysadmin, Audit, Network Security Institute) whose mission is to alert organizations of new exploits as well as notify them where patches or fixes are available to detect and deflect the new exploit or vulnerability. While the model under which these organizations operate supposes that educational, commercial, and governmental entities that encounter new exploits are completely willing to share this information, this may not always be the case.

Often before it becomes widely known that there is a new exploit in the wild, there are rumors that spread through the hacker community about the potential new exploit, spread both by individuals in possession of the exploit as well as those who have heard of the exploit secondhand. Commercial entities often are not aware of the existence of a new exploit for many months. Further, the survival of an exploit is often enhanced by the fact that there is a traditional lack of coordination and cooperation in the commercial sector concerning the sharing of information about recent attacks on their information technology infrastructure. Fearing that any sort of disclosure might eventually become public and damage investors' confidence in the company, many large commercial enterprises refuse to even acknowledge the existence of an attack using some new exploit, let alone share details of the attack with others.

LIFE CYCLE OF AN EXPLOIT

Eventually, the exploit will spread exponentially and end up on countless Web sites being downloaded and deployed by the tens of thousands of script kiddies. At this point, the combination of thousands of individuals using the exploit—as well as the millions of computers on the Internet that lack firewall, intrusion detection system (IDS), or IPS (intrusion protection system) protection—will result in damages to global computing resources that grow not by multiples but by orders of magnitude within a short period of time. Figure 16-3 illustrates this life cycle of an exploit.

Eventually, digital signatures and patches to protect against the exploit become widely distributed throughout the Internet, and the high-value targets that were originally vulnerable to the exploit are now protected. Once it has reached maturity, the exploit may have caused a tremendous amount of monetary and information security damage.

The distribution pattern of a new exploit often looks something like the graph shown in Figure 16-4. During Stage 1, the exploit is tightly held by the original author, and while vague knowledge of its existence may be held by several individuals besides the exploit author, the code is not shared. During the "friends and family" stage (Stage 2), the exploit is shared among a close group of individuals

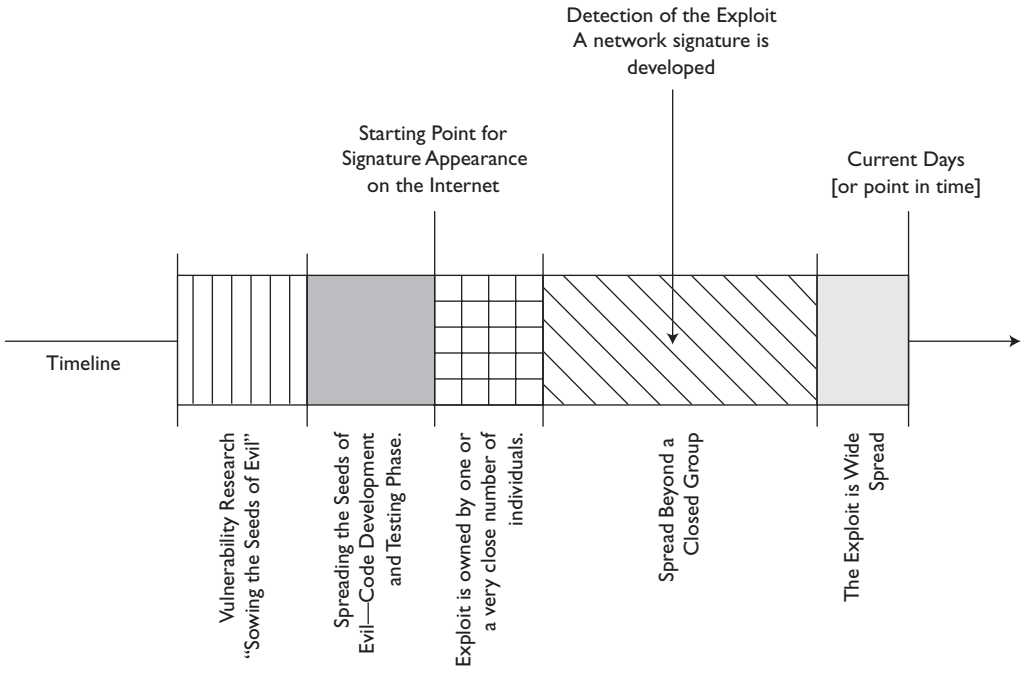

Figure 16-3 The life cycle of an exploit

usually belonging to the same blackhat social group. During this time the exploit may undergo modification and some minor testing.

Stage 3 is characterized by the diffusion of the exploit, usually to a select few individuals who have social connections to one or two members of the original author's hacking group. During the later part of this stage, rumors of a new exploit begin to spread more generally across the blackhat community. Stage 4 is the terminal stage in the distribution of the new exploit where an exponential increase in rate of distribution occurs and widespread deployment of the exploit begins to occur. At this point, the economic and strategic costs of the exploit increase exponentially as well. In order to effectively limit the collateral damage that is done by the exploit, it must be caught prior to the inflection point in the distribution function or significant harm is likely to ensue.

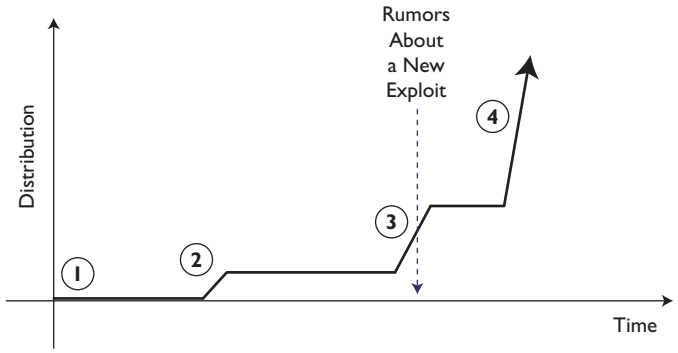

Figure 16-4 The distribution pattern of a new exploit

A DANGEROUS EXCHANGE

As alluded to earlier, once the exploit spreads from the original author or small group, the value of the exploit to the originating group has pretty much been depleted. This fact has not been lost to members of the blackhat community, and it has been known on occasion that when an exploit appears about to leak to the community at large, the exploit is booby-trapped or crippled before it is passed off to others outside the original authoring group. This practice has sometimes resulted in limiting the distribution of an exploit because of suspicions that the exploit might backfire on those using it. It is rumored, for example, that one exploit developed by members of a longstanding blackhat group for the good of the computing community as a whole is in reality spyware, capturing information and relaying it back to an individual or individuals unknown. The always-present potential for this type of double- or triple-cross has had an effect on the distribution of some exploits. The example below illustrates a typical IRC exchange where an exploit is spread from one blackhat to another:

```
.

:_pen :do u have the syntax
:_pen :for
:D1ck :yeah
:_pen :sadmind exploit
```

```
:_pen :?
:D1ck :lol
:D1ck :yes
:_pen :what is it
:D1ck :./sparc -h hostname -c command -s sp [-o offset]
   [-a alignment] [-p]
:_pen : what do i do for -c
:D1ck :heh
:D1ck :u dont know?
:_pen :no
:D1ck :"echo 'ingreslock stream tcp nowait root /bin/sh
   sh -i' >> /tmp/bob ; /usr/sbin/inetd -s /tmp/bob"
```

THE DEATH OF AN EXPLOIT

Exactly when does an exploit die? It is likely that after a number of years there will be systems on the Internet that will not have been patched, upgraded, or otherwise protected against an exploit. Does an exploit *ever* officially die? There may not be a very neat answer to the question of whether an exploit ever officially dies. However, one potential answer lies in evaluating the value of vulnerable systems as an exploit ages. One definition for the demarcation of the death of an exploit might be when the value of systems vulnerable to the exploit becomes nominal—that is, when there is a near-zero value in the systems that could be compromised with the exploit and so there is no motivation to do so. It remains to be seen whether this definition holds up over time.

MEASURING THE RISKS

Finally, a question that often is relevant to companies, organizations, and individuals who are connected to or have a presence on the Internet is, "What are my chances of being the victim of an exploit or attack?" There are several dimensions that factor into this probability. One dimension is the attractiveness of the potential target. A home-based Web server whose sole purpose is to promote the health and welfare of rabbits as house pets has little target value other than perhaps as a launch platform from which to initiate DoS attacks, for example. A more attractive target might be one that has a .gov or .mil domain where one might find interesting information, or perhaps an e-commerce site where credit card information is likely to be stored.

A second dimension that affects target probabilities is the level of visibility of the potential target. Computers that can be identified as belonging to large corporations or governmental entities are often high-visibility sites—the more visibility you have, the more unwanted attention you may attract. A third dimension is represented by Web servers that host Web sites representing particular viewpoints, cultures, religions, political causes, or otherwise contain controversial content—these are often at greater risk of attack from hacktivists.

A third dimension deals with the level of security employed by a potential target. Well-defended information systems protected with firewalls, token systems, and IDSs are less likely to suffer successful attacks from exploits. This is in most part due to the fact that the distribution of exploit authoring skills within the blackhat community is not uniformly distributed—that is, there are a lot fewer very skilled potential exploit writers than there are novice exploit authors, and so the probability of a successful attack on a well-defended system is smaller.

However, even the best-defended systems are vulnerable and have some non-zero probability of being exploited. There is no guaranteed, foolproof protection.[4] Everyone connected to the Internet is exposed, and only through investment in organizational computer security can that risk be attenuated. In terms of measuring risk, the deployment of honeynets as sensor devices gathering data on known and new exploits may be able to assist in the quantification of this risk. In addition, the integration of data gathered through honeynet technology combined with risk assessment methodologies may be useful in producing better, more quantifiable measures of the risk to critical information infrastructures.

4. A humorous anecdote from one of the authors is relevant here. During the process of acquiring an information company, the author discovered that the head of that company's IT department deployed a simple strategy to keep hackers out of their system. His solution was to simply unplug the company network from the Internet at 5 P.M. when he went home at night, citing that hackers only work during the wee hours of the night and early morning. While his reasoning was bad, his solution was quite effective. During the hours of 5 P.M. until 9 A.M. the next morning when he plugged the company's router back into the Internet, his network was reasonably safe from attack. The problem, of course, came the moment he plugged the company router back into the point of presence. His strategy was quickly replaced with a commercial firewall application.

INTELLIGENCE-BASED INFORMATION SECURITY: PROFILING AND MUCH MORE

The art and science of traditional profiling has been around for decades. The ultimate objective that most people associate with traditional profiling is to answer the old Butch Cassidy and the Sundance Kid question, "Who ARE those guys?"—that is, to identify the distinct individual or individuals associated with a particular attack, exploit, or compromise.

Intelligence-based information security is a class of analysis techniques used to determine threats to an enterprise based on current "knowns" about a group—how they operate and what will make them strike next. First we will discuss the types of information we'd like to understand about the person/group in question, followed by the results of an analysis of a group known as Acid Falz. No names are ever disclosed beyond the pseudonyms used by the blackhats themselves in an effort to hide their true identity.

The purpose of intelligence-based information security isn't necessarily to create a 1:1 correlation of blackhat skills to vulnerabilities, but it's amazing to see that the correlation actually exists. Additionally, with so many low- to medium-skilled blackhats using the same tools over and over, profiling several persons or groups in a cross-section of low- to medium-skilled blackhats will offer you insight as to how the vast majority of attempts will be made on your networks.

When looking at a honeynet compromise, or preferably a series of honeynet compromises, there are four main areas of interest you should consider. The analysis isn't all that hard, but keeping it at a high enough level to be able to see beyond the bits and bytes sometimes can be. Remember, to get a feel for anyone, not just blackhats, you've got to try and see something through their eyes. So, when you're looking at a series of attacks, think about the following four things:

- The characteristics of the event
- The consequences of the event
- The characteristics of the blackhat
- The target's characteristics

Some may argue that when taken together, these characteristics may be used to actually predict computer incidents. Let's think about this for a moment. If we know how a specific blackhat likes to operate, we understand his or her typical *modus operandi* (meaning, what they actually like to do once in the system), we know what kinds of things make a blackhat move from thinking about an attack to actually performing the attack, and we understand what kinds of targets the blackhat likes to go for, we can link these series of events together to form a complete picture of the attacker from which a prediction can be made.

CHARACTERISTICS OF THE EVENT

How do we define the characteristics of an event? We need to look at events closely with the idea that something may have happened that caused the attack to take place. What might have caused this particular blackhat to move from thinking about the hack to actually perpetrating the attack? The following are a few of the things to look for in performing this piece of the analysis:

- Was this hack simple vandalism?
- Was there something that precipitated it?
- Was the hack the result of revenge?
- Was this a showing of public support for a particular cause?
- Was the hack purely out of emotional dysfunction?
- Was this hack a challenge hack? Was there an element of status elevation involved?
- Was there an opportunity for either direct or indirect economic advantage, meaning was the hacker stealing money, cards, or information that may involve direct payment from an outside source?
- Was this an attack resulting from a patriotic act or hactivism?
- Was there anything in the news that day that may have hit the political hot button of a particular class of hacker?
- Was this hack used for information warfare? Are we, or is anyone else, moving into a period of increased tensions?
- What about direct military operational or intelligence support? Is it possible that a military event precipitated this attack?

Often we can find the answers to some of these questions through a search of the current day's or week's events, especially when hactivism, political motives, or military action is in the news. Searching news sites on the Web is often a good way to gauge how productive this first dimension of the profiling process is going to be. In this case, it is assumed that there is a nontrivial probability that the actual attack was triggered by some outside event or external motivation.

CONSEQUENCES OF THE EVENT

Next, let's consider the consequences of the event. Did the blackhat consider the consequences of his or her actions? Think about why the blackhat chose this particular target or timing for the attack. Is there any fallout from the attack that may or may not have desired consequences that would benefit the blackhat? For example, the easiest considerations may be that the consequences of the attack may be as simple as the attacker's gaining increased access or bandwidth to carry out further attacks. What about disclosure of information? Corruption of information? Theft of resources, or simply DoS? Every attack has a consequence signature. What was the blackhat's motivation for selecting this one? Was there something to gain, or was it simply a target of opportunity?

CHARACTERISTICS OF THE BLACKHAT

Next, what can we tell about the blackhat himself or herself? This is probably the hardest part of the entire analysis (which makes us very happy we have a Ph.D. on the team who can assist in the analysis of the blackhat psyche!). Some of this is pretty straightforward. The rest is not. It's not an exact science, but get as close as you can and keep records for later use. For ease of use, here's a simple checklist to follow:

- What is the hacker's nationality?
- Is this hacker a member of a group? If so, which one?
- What type of hacker would you consider this to be, based on the skill level of the attack? Possible choices might include military, intelligence, political, terrorist, hactivist, grayhat, corporate intelligence, and so on.

- Consider the amount of knowledge needed to pull off this kind of attack:
  - How much did the hacker need to know about the target operating system to pull off the attack?
  - What about the vulnerabilities or exploit used?
  - How was reconnaissance handled? Quietly? Stealthily? Noisy?
  - Were automated scripts used? How skillful did the attacker need to be to use the tool?
  - Did the blackhat do anything to hide his or her activity? How effective were the activities? Did they simply scrub every log, or were actions taken to conceal the activity through other means?
  - Was there any demonstrated authoring of tools during the attack? In other words, were any tools made specifically for this attack, or were the tools run of the mill?
- What kinds of resources were needed to pull off this attack? In other words, did the blackhat require any special hardware or software to do this attack?
  - How much time was needed for adequate reconnaissance?
  - Was this a low and slow attack, or fast and noisy?
  - Did the blackhat require funding? Was this the work of more than one blackhat?
  - Is there reason to believe the attacker had permission to be on the system (this might indicate an insider threat)?
- Next, let's go back to motivation:
  - Was this hack simply the motivation of a vandal? Simple defacements might be considered vandalism. Is there anything that might indicate this attack to be more than just vandalism?
  - What about revenge? Were there any indicators that might have suggested the attack was in revenge for something else?
  - Was there an indication that the attack might have been used to garner support for a public cause or action?

The answers to these questions help build a profile of the attacker that you can use to accomplish several things. For example, you may have a database of hack-

ers and hacking groups with fields built around these characteristics and so you can, at a simple level, query the database and eliminate those records that do not match the profile. This leaves a subset of individuals and groups that are suspects for the particular attack you are investigating. On a more sophisticated level, you can develop complex mathematical models that generate probabilities of culpability for each of the individuals or groups that exist in your database and focus your investigation on the most probable and work your way from there.

You may also assume that the individual or group responsible for the attack may be someone either new on the scene or a target that has managed to elude your efforts up to this point. In this case, you have a profile from which you can conduct a search for new individuals or groups not in your database that have a high probability of matching your constructed profile.

CHARACTERISTICS OF THE TARGET

Finally, characteristics of the target may also give you some clues as to who is perpetrating the attack. These characteristics can also suggest the profile of other prospective targets within your organization which can be used to identify and notify system administrators to take preventative measures as well as detail what to look for in a possible attack. The following are some questions concerning the characteristics of the target that you should consider:

- What were the operating system and version?
- Which exploit was used?
- Which port was targeted?
- What was the host name?
- What was the IP address?
- Who owned the system? What does that department or company do?
- What's the purpose/configuration of this computer? Was it on the network? Print server? File server? What was the exact function of this machine?
- Does the target company have a security administrator? What is his or her level of experience?

- What kind of information is stored on this computer?
- Who in the company did this computer belong to? Was it the CEO, an engineer, a secretary, a systems administrator?
- Is the owner of this computer a high-profile target? For example, is this a Pentagon target? Microsoft? Would targeting this computer make an impact in the press?
- Would a successful attack on this computer offer access to another? What kind of trust relationships does this computer offer or maintain?
- What kind of network connectivity is attached? Wireless? T? DSL?
- Does this computer require physical access to be attacked?

Using these questions to produce a profile is a time-consuming but often rewarding task. It can not only assist you in identifying the attackers but also help you evaluate where and what is at risk in your enterprise from these individuals or groups. Some blackhats specialize in one particular platform or operating system. Understanding this can help you identify the individual as well as anticipate or eliminate potential targets. It may also be of assistance in configuring a honeynet with that particular platform if it is anticipated that there are likely to be further attacks and scans from this individual or group.

BRINGING IT ALL TOGETHER

Once this mostly technical profile has been developed it can be summarized in a document for distribution and/or further reference. Illustrated below is a detailed profile document on a Russian hacker group named Acid Falz. Notice the magnitude and resolution of detail that can be collected about this real blackhat group.

ACID FALZ

Acid Falz is a group pulled randomly from the Internet that we will use to discuss profiling of groups. No real names have been used other than the pseudonyms used by the blackhats themselves to hide their identities. The following is some background information on the group:

- Group Name: acid fAlz gr0up (acid falz)
- Nationality*:* Russian
- Membership: Six, with two most active (aLph4Num3Ric and Crazy Einstein)
- Favored Targets: 97 percent UNIX and variants, of which 22 percent were Linux
- Favored Methods: CGI vulnerabilities
- Favored Domains: Commercial
- Similarities in Targets: None noted
- E-mail Addresses: *voodoo@acidfalz.ru, crazy_einstein@yahoo.com,* or *alph4num3ric@crackdealer.com*
- Web Site: *http://www.russiahack.com/oldschool.html*

Member Information

Here are some details about several key members of the group gathered from Web sites and list postings:

- **aLph4Num3Ric (primary member):** Self-proclaimed security expert and graphics designer for the site. Also, apparently one of the more experienced hackers in the group, as witnessed by his several published works on the site. The topics include:
  - [Windows] About Windows NT Security
  - [UNIX] About LILO Security
  - [Telecommunications] All About DSL
  - [Misc] Scanners: All About This
  - Port Numbers
- **Crazy Einstein (primary member):** Crazy Einstein is also a self-proclaimed security expert, also experienced in programming and graphics. Crazy Einstein discusses his love for music, which makes us believe he is likely in his late teens. From his published works, it would appear Crazy Einstein sees himself as an application-level programmer and hacker. His published works include:
  - Cracking tutorials (Parts 1–5 and cracking games), including lessons in cracking WinZip, mIRC, and Secret Agent
  - Principles of Cryptography

  - Script Security
  - Self Security
  - Protecting Programs from Cracking
- **DangerDuo:** Very little information exists about this fairly new member. However, DangerDuo appears to be a Win32 programmer/hacker, which would explain the two Windows defacements in March and July 2001.
- **Parad0x:** Parad0x sees himself as a "Security Expert" (don't they all). Parad0x does not seem to be an active participant.
- **RED:** Unknown. Again, RED does not appear to be active with the group.

Tools Written by Members of Acid Falz

Note that Crazy Einstein wrote all but one tool. DangerDuo, who is listed as a new member, wrote the other, written for scanning Win32. This might explain the two Windows hacks shown in the defacement tracking site alldas.de (this Web site is now gone).

- **aUto dEface tool (Crazy Einstein):** Auto-defacement tool
- **Port scanner for Win32 (DangerDuo):** Port scanner for Win32
- **Script Analisatov v1.0b (Crazy Einstein):** Program for search bugs in CGI programs
- **aCid fAlz Intruder v1.0 (Crazy Einstein):** Program for Gamers: modify save files
- **aCid fAlz CGI Scanner v1.0 (Crazy Einstein):** Scanner of CGI bugs with lots of options
- **aCid fAlz FTP Brute-Force Attack v1.0(Crazy Einstein):** Easy FTP brute-force program
- **aCid fAlz WordList Creator v1.0 (Crazy Einstein):** Program for creating word lists to a program that cracks passwords
- **aCid fAlz WordList Creator: MUTATION: v1.0 (Crazy Einstein):** Add-on to WordList Creator with some special options—you may create word lists from any file
- **aCid fAlz Port Scanner v1.0 (Crazy Einstein):** Simple port scanner for UNIX

- **aCid fAlz POP3 Brute-Force Attack v1.0 (Crazy Einstein):** Easy POP3 brute-force program
- **aCid fAlz XXX Brute-Force Attack v1.0 (Crazy Einstein):** Easy XXX Sites brute-force program

IRC PROFILING: ANOTHER VIEW

One of the more productive avenues in traditional profiling turns out to be in the area of profiling using information gathered from IRC chats. There are several fundamental reasons behind this. Sometimes we forget that behind all the technology—the computers, the physical layers of the global data communications network, the network protocols, and the software code—there exist human beings; these inherently social creatures driven by social forces and motivations. Consistent with basic human nature is the desire to form social groups and communicate with others within these groups. The lack of geo-propinquity present in these social groups is encouraged by the fact that computer technology facilitates communication at a distance, and so many blackhat groups consist of individuals who live often thousands of miles away from each other. IRC chat is one of the more immediate and popular communications channels for pretty much the entire spectrum of the color-hatted community. It is popular because it possesses attractive qualities to these individuals such as *n*-way communication and the ability to communicate in near-real time using the technology that has enraptured them.

One trend that is reducing this channel of intelligence is the growing use of encrypted IRC communications channels such as is the case where blackhats are using SILC (Secure Internet Live Conferencing) clients and servers. Encrypted IRC channels make the process of gathering information about blackhats more difficult. However, it also significantly alters the IRC environment for these individuals. Individuals communicating on encrypted channels understandably feel a stronger sense of privacy and protection and are therefore much more likely to disclose sensitive information. This greatly enhances the potential value of information that would be gathered by an individual with covert intentions who has gained the trust of the group and exchanged the necessary keys to join, monitor, and contribute to an encrypted SILC chat channel.

As any good profiler will tell you, where there is communication, there is information, and information can be mined, stored, sorted, sifted, and analyzed. Human dialog is an extremely rich communication channel. Face-to-face communication is likely the richest variant of human communication, containing high bandwidth verbal and nonverbal flows of information in the form of facial expressions, speech rate, eye gaze behavior, not to mention the actual content of the communication itself.

Unfortunately, as noted earlier in the chapter, much of this bandwidth is lost in IRC communication, a fact that IRC participants themselves often realize. Therefore they often use communications crutches such as emoticons to "widen the bandwidth" a bit. However, there is a tremendous amount of information that can be extracted from communications like IRC chats that is useful to the traditional profiler. In the discussion to follow, we'll use several examples of real IRC communications to illustrate important points. These are taken from a series of IRC sessions legally obtained from a computer within a honeynet that had been compromised by a particular group within the computer community. Only the "handles" or nicknames of the individuals involved in the chats and the exact hours of the communication are altered to cloak their identity, although it is likely that individuals reading this chapter that were involved in the discussions will recognize the exchanges.

This first excerpt is unusual in that some of the individuals in the chat disclose their general geographic location in the clear:

```
21:59:16 shaverboy: checkov, where in the us are you?
21:59:16 quark: lol
21:59:17 checkov: find me a better hobby
21:59:23 checkov: shaverboy: NY
21:59:30 quark: Maine here
21:59:47 checkov: quark: I was in maine like a month ago
21:59:51 checkov: it sucks there
21:59:55 quark: lol, yeah it does
22:00:02 shaverboy: i was born in maine
22:00:03 checkov: wtf is up with all the ice
22:00:08 burgerking: ACTION puts on his l337 glassess
22:00:11 quark: Note: Never, under any circumstances, move to maine
22:00:19 checkov: there is like rocks of ice on the sidewalks
22:00:22 shaverboy: checkov i'm in VT, just got 2 feet of snow on x-mas day
22:00:24 shaverboy: i love maine
22:00:25 quark: lol
```

```
22:00:30 checkov: i hate snow
22:00:36 checkov: I lived in fl for 15yrs
22:00:40 quark: snow sux0rs unless you're skying on it
22:00:41 shaverboy: dang
22:00:47 quark: err
22:00:50 quark: skiing
22:00:52 burgerking: skiing
22:00:53 burgerking: lol
22:00:56 quark: lol
22:01:05 quark: I was like... wait...
22:01:39 burgerking: shaverboy did you get the pre built module one?
22:01:52 burgerking: *vmware
22:01:52 shaverboy: no i don't think so
22:02:00 burgerking: what one you get?
22:02:32 quark: so yeah, I woke up at 6:30 am to get ready for
    what I thought was an orthodontist apointment... turns out
    it was at 3:40 in the afternoon
22:02:38 quark: I could have slept in too :(
```

Not only do we have states of residence for some of the individuals, one of the group has an orthodontist appointment on a specific date at a specific time. Further inspection of conversations uncovered the fact that this individual lived in a small town in Maine. Given the proper motivation, legal justification, and law enforcement credentials, this individual could be easily found.

This second excerpt gives the profiler an idea of the skills, platform, and languages of choice that might be used in an exploit if these individuals were blackhats:

```
20:49:30 quark: am I the only one who uses C++ rather than C?
20:49:32 oracle: heh
20:49:34 shaverboy: yah
20:49:42 oracle: u a winshit coder?
20:49:42 shaverboy: personally i don't like c++
20:49:42 burgerking: outties
20:49:49 burgerking: ".k *"
20:49:52 quark: lol, yes, i'm a winshit coder
20:49:52 burgerking: .users
20:49:59 shaverboy: i can do everything i want in C and if i
    need object oriented stuff, I can use LISP, Java or Python
```

These kinds of discussions are common in IRC chats. Individuals are often attempting to evaluate their status within their group and one way that is done is

to compare skills in different areas such as programming languages and platforms. The next excerpt allows the profiler to identify the status positions of at least some of the members of the group:

```
15:34:36 checkov: what code is that?
15:34:42 burgerking: lol checkov its impossible to help him...
    Slash how old are you?
15:34:46 Slash: 14
15:35:02 burgerking: same as ashran :)
15:35:16 Slash: and burgerking shhh!!! i am trying to learn
15:35:26 quark: then there's the different logos I made for
    www.texas-lamers.com
15:35:28 Slash: checkov i am not sure what kind of code it is
15:35:46 cigquake: because you don't know shit about what is going on
15:35:50 burgerking: yeah quark im just an amature :P
15:36:09 quark: lol, I'm far from pro, I just enjoy doing it
15:36:17 checkov: Slash: well figure it out
15:36:36 burgerking: Slash the whole point of me pestering you
    is so you will get off your ass and try learn.. because you
    rely on others
15:36:46 burgerking: and thats not what your suppose to do to learn
15:37:01 Slash: i am learning i never learnd why !/bin/pass workes!!!
16:34:04 burgerking: Ok well here is a simple explanation the
    code your exploiting has a group level of 2.. which is your
    current the user is level3 which means
16:34:13 quark: this is the first logo I ever made
16:34:17 burgerking: quark stop the logo spam :p
16:34:26 quark: lol
16:34:28 Slash: that he holds the pass to level3?
16:34:30 quark: last one :)
16:34:47 Paris: quark: you prolly shouldnt post links that have
    your real name on them
16:34:51 burgerking: No it gives you elevated privliges to
    level3 so you can view the /bin/pass at level3
16:34:52 Paris: or your pic ;\
16:35:00 Slash: o
16:40:32 burgerking: is Slash still peon?
16:40:44 Slash: yup
16:40:55 Slash: i really have no use for op
16:40:58 burgerking: do you know what peon is
16:41:02 Slash: nope
16:41:24 burgerking: means you can't speak unless someone gives
    you the right to spea
16:41:25 burgerking: k
16:41:32 Slash: ok
```

It's obvious here that Slash occupies most likely the lowest status position in the group. His lack of experience and the admonition that he needs to learn on his own are consistent with the hacker culture's emphasis on self-learned skills. Slash's status position is uniquely formalized by the peon property assigned to him in this chat room. The peon property prevents him from speaking without first getting someone of higher status to grant him the opportunity to contribute in the chat room.

Status processes also often lead to other disclosures of information. That is, often, information about status characteristics such as age, gender, education, employment status, and race are exchanged. These characteristics are also quite useful to the profiler in terms of identifying the individuals involved. Here is an excerpt where not only do you learn ages of individuals in the group but also many of their nicknames:

```
21:08:47 shaverboy: how old are u?
21:09:14 quark: 17
21:09:16 quark: lol
21:09:28 quark: and your self?
21:09:34 Criminaljustice: ACTION is away: lunch
21:11:03 shaverboy: i'll be 23 in a month
21:13:42 quark: lol, I'm guessing i'm fairly young in comparison
    to most of the people here
21:14:10 shaverboy: nope
21:14:12 shaverboy: i'm the old guy
21:14:17 quark: lol
21:14:25 shaverboy: other than fastburn who's 25 and varied1
    whos a little older
21:14:26 shaverboy: temple is like 14
21:14:30 shaverboy: most of the people are 17 - 18
```

Often you can classify individuals within the group in terms of whitehat, grayhat, or blackhat by analyzing activities that the individuals say they participate in. However, in some cases, people self-identify themselves, although it is not unusual for individuals to exaggerate their exploits. Here is another excerpt that reveals one of the members suggesting that they are in fact blackhats:

```
16:44:56 Shortkid: i used to be gray but its not that cool
16:44:59 burgerking: Ashran im not from the south island ;)
16:45:01 shaverboy: black hat eh?
16:45:15 burgerking: lol how are you a black hat?
```

```
16:45:15 shaverboy: so you're actually trying to be malicious? that's fine by me
16:45:32 Shortkid: lets say i want to be a black hat
16:45:37 shaverboy: ok
```

Burgerking's reference to the South Island refers to the fact that he lives in New Zealand, as he reveals in later chat sessions. It is not unusual to have individuals in different countries belong to the same social hacking group. Often when these cross-national groups chat, the discussions are more productive in terms of information that leads to specific geographical locations. This is because the individuals in these groups end up using and subsequently explaining terms and concepts native to their national origin and culture while unfamiliar to the rest of the members of the social group.

This discussion only touches the surface of many of the techniques used in profiling individuals using inter-group communication. There are many principles of profiling in use today that are beyond the scope of this book and so are left to the reader to pursue in more depth.

SUMMARY

This chapter examined three areas of interest to those engaged in the areas of computer security: profiling, understanding the motives of the community, and the life cycle of an exploit. It is hoped that the discussions presented here will assist the computer security community in better understanding the environment and forces that shape their everyday lives. The hacking community as a whole is a rapidly evolving entity and so the reader is encouraged to take the points made here as starting points for making sense of the present and future course of computing technology and its relationship to a secure computing environment.