

1

Responding to Attacks

It's Saturday night. Your network is well designed, well run, and well supported. Your security team is well trained and your policies and procedures are committed to paper. But in the rush to get the policies and procedures out the door on time (so you could get that manager's fat bonus check), you forgot to include incident-response procedures. And while you're congratulating yourself on a job well done, a hacker breaks into your most critical system.

Now what? How quickly (and whether) you can answer that question could determine the fate of your data. Employees need to know what to do, how, and when. They also need to know to whom to report the break-in. Otherwise, the situation can get out of hand quickly. Proper escalation is especially important if the scale of the break-in goes beyond your support team's knowledge base.

When a break-in occurs, every move you make can mean the difference between saving or losing your company secrets. Just imagine what would happen if all the essential information on your computer system were stolen or destroyed. Unlikely? Sounds unlikely to most people until it hits their systems!

Remember, the data on your network is important! So, be prepared. Make sure everyone (from the top down) in your company knows what to do in the event of a break-in to save your data from theft, modification, or destruction. Just consider ...

Incident-Response Nightmare

Dave Armstrong was a system administrator supporting the intranet for First Fidelity Bank of Anacast County. Late one Monday night, Dave watched as a hacker gained full control of all 200+ systems and began wandering through them at will, collecting passwords and perusing data.

Unfortunately, Dave did nothing but watch as he tried to figure out who was on his system in the middle of the night. Although First Fidelity had written policies and proce-

dures for most other situations, there weren't any formal incident-response guidelines. Because Dave had no specific instructions, he spent a full three days trying to identify the hacker—without success—before escalating the call to the bank's security team.

Just imagine, for a moment, a hacker roaming unchecked through your own bank's network for three days, collecting names and account numbers, possibly even modifying data, transferring funds, or destroying records. Thinking about changing banks? I would be!

How does a situation like this arise? In this case, Dave configured a software server so that it was trusted by the other systems. Trust in this sense meant that all the systems on the network were granted remote root access to the software server without first requiring a password (a web-of-trust among systems). Several hundred systems trusted the software server.

Although this arrangement makes it easy to distribute new software, it can be risky, especially when the risk and vulnerabilities associated with supporting trust are not clearly understood in the first place. If a system *must* be configured as a trusted server (no other practical options can be applied), the trusted server *absolutely must* be secured. Otherwise, any hacker who breaks into the trusted server has immediate root access—no password required—to *every* system that trusts that server.

That's what happened on First Fidelity's intranet. Hundreds of systems in the intranet trusted the software server. As a result, the server provided a tempting target for any hacker seeking entry into the bank's computer network. Dave had no idea that the system was at risk and unable to withstand attack. It never occurred to him (or his manager) that a single unsecured system would open the door to the rest of the network.

For First Fidelity, the web-of-trust was spun far into the depths of their intranet (200+ systems). With hundreds of systems trusting the software server, the server should have been protected with proper security controls. The server, however, was lacking security altogether. It was wide open, just waiting for a hacker to walk right in.

And, that's exactly what happened. When the hacker gained full access to the trusted server, remote root access to all the systems on the network was granted. This hacker didn't have to work very hard to gain control of the entire network.

Let's take a closer look at the details of this break-in and what happened during the days that followed.

Day 1: Unauthorized Access

Dave discovered the hacker's presence at 11:45 Monday night, while doing a routine check of the network. He noticed that some unusual processes were running and that CPU utilization was much higher than normal for such a late hour. This unusual activity sparked Dave's curiosity, so he investigated further. By checking logins, he discovered that Mike Nelson, a member of the bank's security team, was logged onto the system. Mike was a legitimate user, but he shouldn't have logged on without first alerting someone in Dave's group. Was this a hacker masquerading as Mike? Or, was it Mike working on a security problem? If it was Mike, had he forgotten about the prior-notification proto-

col, or had he deliberately neglected to notify anyone? Dave had no idea. Even worse, he had no idea who to call or what to do.

What happened next? The same thing that happens to most people the first time they suspect a hacker has broken into their system. Dave experienced a rush of adrenaline, a sense of excitement mixed with fear, and confusion about what kind of action to take. He was alone. It was the middle of the night. If he hadn't been working late, it's possible no one would ever have known of this attack. He decided that since he was responsible for the system, he should do something to regain control. He kicked the user off the system, then rendered the account useless by disabling the user's password. Dave had control of the system again. Thinking his mission was accomplished, Dave went home.

Unfortunately, Dave didn't realize that action was a short-term response to the situation. Kicking an unauthorized user off the system often means *merely* that he's off for the day. It doesn't mean he won't be back. Once a hacker gets into a system, he often leaves back doors that allow for easy access next time. Dave's action left him with a false sense of security. Dave assumed that he had solved the problem by simply throwing the hacker off the system. But, the security problem that let the hacker on in the first place had not been addressed. Dave may have scared the burglar out of the house, but the door was still unlocked.

Day 2: Problem Fixed

Tuesday morning, Dave described his midnight adventure to his manager and two other system administrators. They discussed the incident for a while, but still had no idea whether the system had been invaded by an unknown hacker or by Mike from security. At any rate, they considered the problem fixed—the suspect account had been disabled, and there were no new unauthorized users on the system. So, they dropped the issue and went back to work. As on most support days, time flew by.

At the end of his shift, Dave logged into the software server just because he was curious. He noticed only one other login, from Ed Begins, the system administrator who ran backups on the servers at night. That seemed normal, even expected. The system was running fine. So, with another 12-hour day under his belt, Dave logged out and went home.

Day 3: Security Is Breached Again

Dave slept in. After all, it was only Wednesday morning and he had already worked 24 hours that week. When he returned to the office that afternoon, he noticed that Ed hadn't logged out of the server the night before. That was odd. Ed worked the graveyard shift and wasn't usually around during the day. Given the unexplained login from Monday, Dave paged Ed to verify his activity on the system. Ed responded to the page instantly. He informed Dave that he had not run any backups the night before, and he wasn't using the system currently. It began to look as though a hacker was masquerading as Ed.

Upon further investigation, Dave discovered that the phony “Ed” was coming from Mike’s system. What’s more, this user was not only checking to see who else was logged on, but also running a password sniffer. Dave thought that Mike was playing around on the system and currently had access to the system by masquerading as Ed. (Dave never seriously considered the possibility that there was an unknown hacker on his system stealing data.) Dave was seriously annoyed by now. He figured that Mike was causing him to run around in circles and waste his time. Dave’s tolerance level was low. He kicked “Ed” off the system, disabled his password, and reported the new development to his manager.

The manager called Mike to ask if he was logged onto the system and using a password sniffer, and to question him about Monday night’s activities. Mike emphatically insisted that the mysterious user was *not* him. Mike also claimed that no hacker could have logged onto his system because he was certain it hadn’t been compromised. Mike’s opinion was that the hacker must be spoofing—that is, pretending to come from Mike’s system but actually originating from somewhere else.

At this point, the situation degenerated to finger-pointing. The system administrators continued to believe that Mike was playing around on the network. Mike continued to insist that the break-in was a spoof and that he was being falsely accused. Everyone lost sleep and wasted more time trying to pin down what had actually happened.

Days 4 to 7: Escalating the Incident

On Thursday, Dave’s manager escalated the problem to the bank’s security manager and the internal audit department. Several days went by while all parties—the security team, the audit department, and the system administrators—waited for the hacker to reappear.

But the hacker never came back. The internal audit manager was left wondering if there had really been a hacker in the first place. Did kicking him off the system a couple of times discourage any further attacks? Had Mike been hacking around for the fun of it and stopped when he realized that everyone was on to him?

Day 8: Too Late to Gain Evidence

A full week after the break-in, the internal audit department contacted Dave and asked for the technical data he had captured that demonstrated the hacker’s activity on the server. Since the bank didn’t have a security expert on staff, the audit manager hired me. My job was to review the technical data and determine who broke into the server.

Day 9: Who Was the Bad Guy?

When I arrived, I discussed the case with the audit manager and reviewed the data. Several days had passed since the second break-in, and the hacker had never returned. Unfortunately, I couldn’t provide the answer the auditor was looking for, because it was impossible to trace the hacker using the data they had gathered. The information did tell me that the intruder had used a free hacking tool (esniff) that is easily available on the

Internet, masqueraded as several legitimate system users, gathered a bunch of passwords, and appeared to be coming from Mike's system. But there wasn't enough data to tell whether the hacker was an outsider, Mike, or someone else in the company.

When Dave kicked Mike off the system, there was no way to trace back to the source. Any answer I gave would have been pure guesswork. Interviewing the staff wasn't helpful. Plenty of fingers pointed to Mike, but no one had any evidence. Lacking that, the best I could do was advise the audit manager to have the company develop and implement incident-response procedures right away.

If it was a hacker, it was possible that back doors into the system were left behind. In the corporate world, a week might not seem very long. But in investigating the scene of a computer crime (yes, breaking into systems is a crime!), it's an eternity. When so much time passes between a break-in and an audit, valuable information is modified, lost, and sometimes impossible to track.

I pointed out that the break-in was made possible by the lack of security on the trusted software server, and that the vulnerabilities needed to be corrected. Furthermore, it was impossible to know how the hacker broke into the server, because there were several vulnerabilities the hacker could have exploited to gain root access. Old password accounts existed, excessive file permissions existed, security patches weren't installed, and so on. The hacker had his pick of approaches.

I told the audit manager that the facts were staring everyone in the face. One unsecured trusted server opened up the entire network. Since the system could have been breached by a real hacker, Dave needed to reinstall the server, add adequate security controls to protect the server, and consider other technical solutions for updating software on their intranet.

I also discussed with the auditor the importance of having a security team you can trust, focusing on the need to thoroughly screen security personnel before hiring. I explained that proper procedures for the security team to follow should be in place, and that all employees should be expected to follow those procedures. Just because they are members of a top-notch security team doesn't mean that they should be able to roam around all of the systems without proper notification. In this case, since a security team member was a suspected culprit, it would have been helpful to have a procedure in place for routing the investigation around the security team to higher management. This type of contingency should be covered under the conflict-of-interest section in the incident-response policy.

Summary: Attacks from the Inside

These two break-ins caused a number of bank staff members to spend a lot of their work time investigating the hacker problem instead of doing their actual jobs. Dave took the problem into his own hands and made important decisions that could have placed the systems and data on his network at risk. He also decided that he was dealing with Mike from the security group without proper evidence to back up his accusation.

Although we'll never know whether Dave was right or wrong in accusing Mike, he was definitely right to recognize that hackers can come from within your network as well as from the outside. As Figure 1-1 clearly illustrates, insiders are a serious risk. Of course, knowing that insiders are a risk and doing something about it are two different things. To protect your data, you need policies, procedures, and training. To many employers, protecting data from their own employees sounds ludicrous. Just remember to look at the 1's and 0's in that data as real money (\$\$\$). Banks don't think twice about implementing adequate controls on the storage of cash. For example, they don't leave the safe wide open so that anyone who works for the bank or any customer who strolls into the bank can walk in and take some of that cash. When data is considered to have the same value as money, security controls on data become a requirement, not an afterthought.

This time, First Fidelity was lucky. With unrestricted access to the network for three days, the hacker could have destroyed data, shut down systems, or even changed hardware setups. Part or all of the network could have been rendered useless. System administrators could have faced days or even weeks of work just getting the systems running again—assuming that current backups existed.

Hackers can cover their tracks quickly, making it very difficult and far too often impossible to trace them back to their starting points. If you don't act right away, you may never even know if data was stolen, modified, or destroyed. For this reason alone, anyone who owns and maintains a computer network must develop clear, specific incident-response procedures.

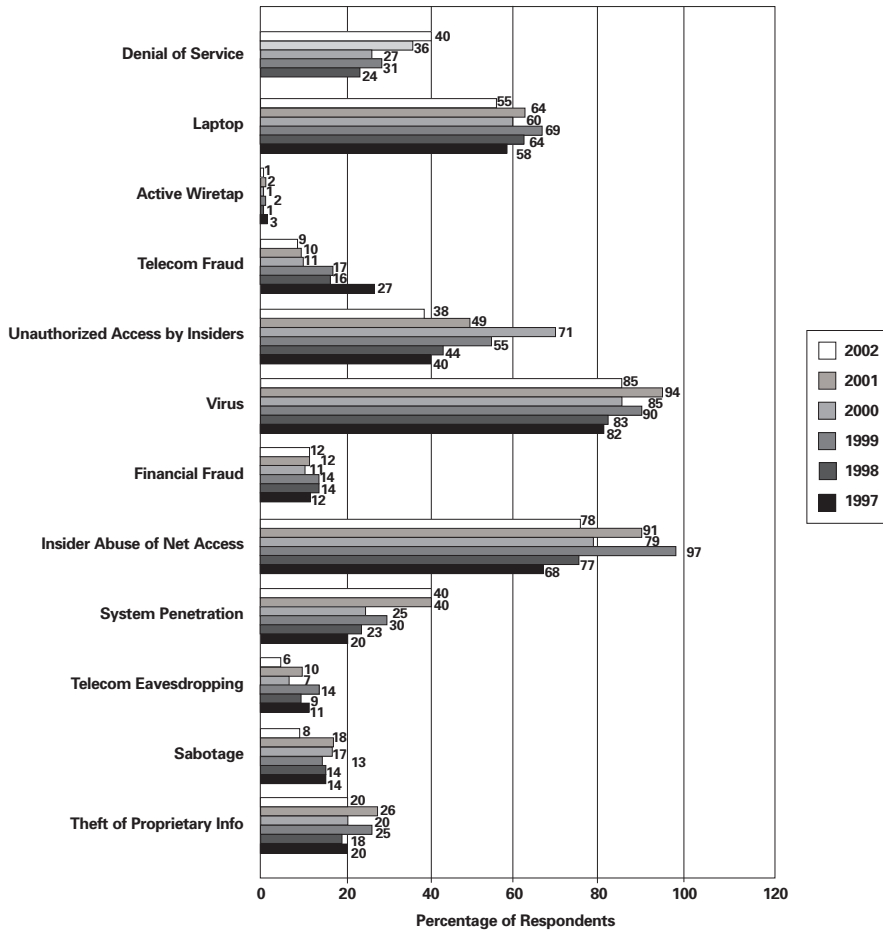
Let's Not Go There...

Given the sensitive nature of their data, First Fidelity was lucky. Of course, relying on luck is not a good security approach. Here's what they should have done instead.

Focus on Prevention

Given the alternatives, you're probably wondering why First Fidelity used such a vulnerable configuration. After all, why expose your data to that much risk? The answer, of course, is, "Why not?" After all, there was no way that a hacker could break into *their* system. Surprisingly, a large number of companies still think this way.

Types of Attack or Misuse Detected in the Last 12 Months (by percent)



CSI/FBI 2002 Computer Crime and Security Survey
 Source: Computer Security Institute

2002: 455 Respondents/90%
 2001: 484 Respondents/91%
 2000: 583 Respondents/90%
 1999: 460 Respondents/88%
 1998: 428 Respondents/83%
 1997: 503 Respondents/89%

Figure 1-1

Don't Think, "It Can't Happen to Me"

Many companies honestly don't see computer break-ins as any more likely than hitting the Lotto. Since these companies are so sure that they are somehow immune from hacker access, they don't take even basic precautions. Since it will *never* happen to them, they

don't budget for security. They don't develop incident-response procedures. Therefore, they don't train their staff on how to respond to an incident.

Simple as it sounds, the most important thing you can do to prevent a break-in is to realize that ***it could happen to you!*** To prevent it from happening, use your most effective security tool—training. Train everyone! From the highest-level manager to the lowest-level data-entry clerk, everyone should know how to protect data from theft, modification, or destruction by unauthorized users. After all, a malicious hacker with too much access could put everyone out of a job!

In a strict sense, unauthorized use is any use of the computer system not specifically authorized by the system administrator(s). Thus, an unauthorized user could be a malicious hacker, a cyber-joyrider, or even an employee who isn't allowed to use a particular system at a certain time or for a certain purpose. In the incident at First Fidelity, the unauthorized user detected could have been any of the above.

As the Computer Security Institute (CSI) discovered in a recent survey, and as illustrated in Figure 1–2, far too many managers are unaware of just how pervasive unauthorized access or misuse is.

WWW Site Incidents: What Type of Unauthorized Access or Misuse?

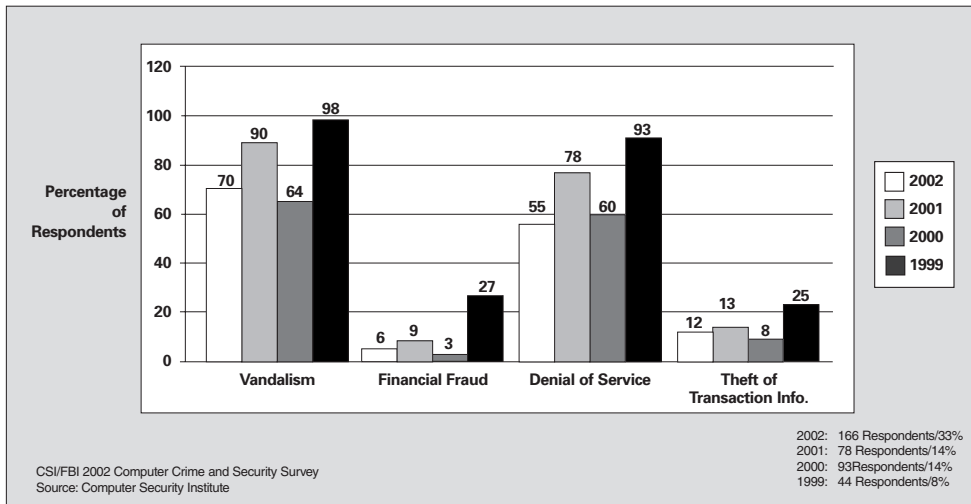


Figure 1–2

Know When You're Under Attack

The first problem in handling a break-in is to recognize when your system is being broken into! You need to be sure that what you're seeing is a real break-in and not just a hardware or software quirk or bizarre user behavior. Detection software may help to determine if your system is under attack in the first place. Installing detection software before an attack is absolutely critical, however. Consider the recent impact of Code Red. On July 19, 2001, Code Red infected 359,104 hosts, which were compromised in only 13 hours. At its peak it actually compromised some 2,000 new sites a minute, even sites that had detection software installed.

Most Intrusion-Detection Systems (IDSs) can detect the attack only if a signature exists. Sounds silly if you think about it. It's like waiting for a burglar to break into your house before you purchase a lock for the door. Furthermore, once you have a signature installed it is easy for adversaries to launch a new version of the attack and slip by the IDS.

Make sure your IDS can detect new zero-day attacks (sometimes called first-strike, or unknown attacks because they have not yet been reported, they are not publicly known, and signatures do not exist). If your IDS cannot detect zero-day attacks you need to update your architecture. Doing so helps you to protect against attacks that target protocols, like Code Red and Nimda and their variants.

I'm not suggesting that you install detection software on every system on your network. Strategically installing it in key locations (on networks and mission-critical systems), however, can give you the upper hand.

Prepare for the Worst

Although prevention is 80 percent of most cures, there is always the other 20 percent. The truth is that no matter how well you plan, there are always unforeseen problems. Being able to deal with those problems often boils down to having prepared for the unknown. To avoid the situation that First Fidelity found itself in, do the following:

Develop a Written Policy for Dealing with Break-ins

If your company lacks a written policy for dealing with network intrusions, you're not alone. Although we tend to focus on large U.S. companies, slack security extends beyond national borders. A 2001 survey of Canadian firms conducted by KPMG found that only half of the respondents had incident-response procedures for handling e-commerce security breaches.

Would Your Organization Consider Hiring Reformed Hackers as Consultants?

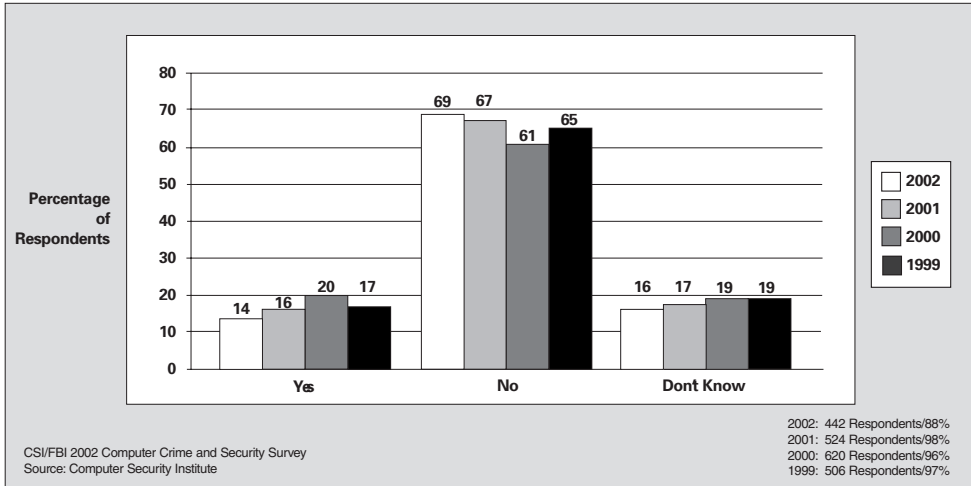


Figure 1-3

Hire an Expert If You Need One

Forming an Incident-Response Team (IRT), developing policies and procedures, and keeping everything up-to-date can be a huge task. It takes time, knowledge, and coordination of staff and resources. If you don't have procedures in place and no one in your company has the expertise to develop them, hire an expert. And "expert" does not translate to "hacker." Be careful whom you hire. As illustrated in Figure 1-3, most companies do not hire reformed hackers as consultants.

There are several companies that take this issue seriously and can provide valuable services. (For details, see Appendix A, "People and Products to Know.")

While developing incident-response procedures for a company several years ago, I talked with an executive from a security consulting company about the backup security expertise they provide. I asked how long it would take if we needed an expert onsite. "We have worldwide coverage," he said, "and can have a team onsite anywhere in the world in minutes to hours, depending on the location." Security companies that offer this type of service are ready and willing to assist, and will send their experts to you immediately if a problem surfaces. They have seen disasters, and they know how difficult it can be to clean up after a serious break-in. It's important to build this kind of relationship before you have a break-in, so that you know someone will be able to respond to your call if, or when, you find yourself in the midst of a disaster.

Get (or Provide) the Needed Training

Even when incident-response procedures exist, system administrators and users may not have been trained in their use. Policies and procedures that aren't clearly understood are not very useful. They may even give everyone a false sense of security. Not only do emergency procedures need to be well documented and distributed, but every computer user in the company—from the Chief Executive Officer (CEO) to the data entry clerk—needs to know how to implement them. Responsibility for computer security falls on every employee's shoulders.

It's a good idea to test your policies and procedures before an incident occurs. Consider a dry run. You may even want to hire a penetration team to test the security of your site. The Tiger Team can try to break into your site and at the same time test your team's response to a break-in. It's not a good idea to leave people guessing whether this is a real break-in or not, however. In other words, don't cry wolf. If you hire a security consultant to test your site security and response to a break-in, inform the support staff. Let them know that this is a dry run and not the real thing.

Designate a Point of Contact (POC)

During a break-in, the clock never stops ticking. If you have to think about who to call or what to do, precious time slips away. Procedures should designate who needs to be notified of the break-in. Your company should have a Point of Contact (POC)—the equivalent of a 911 emergency line—that users can call in the event of a break-in.

Understand and Prioritize Your Goals

Your company goals and priorities may be different than those of the guy next door. The bottom line here is that complex incidents don't allow time to think about the priorities. Therefore, your goals during a break-in must already be documented and understood before the break-in occurs.

Knowing your goals is essential to formulating an appropriate plan of attack. The goals appropriate for your network may include some or all of the following:

Protect customer information. You might maintain critical customer information on your network. If a hacker steals, modifies, destroys, or even posts the information to the Internet, you may find yourself in court.

Contain the attack. Prevent the use of your systems to launch attacks against other companies. Sometimes you may need to disconnect a system from the network to prevent further damage and limit the extent of the attack. For example, if you have a customer network (extranet) connected to your network and a hacker obtains access to the system that connects you to your customer's extranet, you must protect your customer's network. If you have to, be prepared (and know how) to pull the plug.

Notify senior management. Management is responsible for the adequacy, accuracy, and reliability of data. If the systems in your company are being broken into, the Chief Information Officer (CIO) should be informed and kept abreast of the situation.

Document the event. Recording all the details may provide management with the information necessary to assess the break-in and could assist in the prosecution of specific individuals.

Take a snapshot of the system. A snapshot is basically a photograph of what a computer's memory (primary storage, specific registers, etc.) contains at a specific point in time. (Sometimes, a snapshot is called a system dump.) Like a photograph, a snapshot can be used to catch intruders by recording information that the hacker may erase before the attack is completed or repelled. As such, a system snapshot provides crucial audit information.

Contact a Computer Security Incident Response Team (CSIRT). It's important to contact a CSIRT (e.g., CERT) during the early stage of the intrusion, because they may have information that can help you bring your intrusion to a close. For example, they may know how to fix the flaw in the vendor's software or hardware that allowed the intruder to access your network. They also compile statistics regarding the total number of break-ins and techniques used by hackers to gain entry. If you have your break-in under control and have fixed the problem that allowed the hacker to gain entry, you should still contact a CSIRT so they can keep accurate statistics. They will not share your company name or tell anyone that you were broken into. Many CSIRTs exist around the globe. For details, see Appendix A, "People and Products to Know."

Identify the intruder. This entry seems obvious, but it isn't always at the top of a list of priorities. Sure, it's nice to get even. But, it's even more important to get by. Don't get so caught up in trying to catch the intruder that you compromise the integrity of your data. If you cannot easily trace back an attack on your own network, you need to consider how important it is to be able to do so. Some vendors offer software that can easily track back attacks (as long as software is installed upstream). This should be strategized at the executive level of the organization.

Know who's responsible for what. Having clear-cut responsibilities removes any ambiguity that can arise. Knowing who's responsible for what facilitates speed and increases the likelihood of identifying the culprit.

Know whom you can trust. The actual break-in was only part of the real problem at First Fidelity. The other part was a lack of trust between key players. If we assume that Mike was guilty, the trust issue becomes a personnel problem. Were appropriate background checks conducted? As much as it seems an invasion of privacy, a thorough background check is essential for anyone who will be responsible for computer security.

If we assume that Mike was innocent, the trust issue resurfaces as a communications problem. Why didn't anyone call Mike early on? Was Dave uncomfortable speaking to Mike because Mike was from security? A phone call could have opened up communication channels and perhaps avoided the finger-pointing that ended up obscuring the investigation. Perhaps there was a history of unspoken mistrust between the system administrators and security team. Employee resentment or mistrust of the company's security team is a serious issue that needs direct attention. Ignoring such a problem puts the company at risk. A procedure for handling a conflict of interest would also have

helped Dave. He would have been able to sidestep the security team by escalating the investigation to a higher level of authority.

React Quickly and Decisively

As the T-shirt so eloquently puts it, “stuff” happens. So, if a hacker breaks into your system in spite of your thorough safeguards, take at least the following measures.

Act Quicky!

The surest truth in security is that the longer you take to react, the more likely it is that the intruder will escape unharmed—with your data—unidentified and prepared to strike again later.

Follow the Game Plan

The whole point of having an incident-response procedure drawn up in advance is so that you (or your staff) can react immediately without having to think about it. Don't second-guess that plan—just do it!

Record Everything!

Once a system is suspected to be under attack, it's extremely important to obtain information. Take a snapshot of the system. Any audit information you can gather is valuable and may ultimately help identify the source of the attack and prosecute the intruder.

Escalate the Problem When Necessary

Escalation is the referral of the problem to a higher level of authority. The incident-response procedure should indicate under what circumstances the problem should be escalated both internally and externally.

Internal escalation—the referral of the problem to a higher level of authority within the company—is required whenever the scale of a break-in goes beyond the knowledge base of the support team. External escalation—calling in an outside expert—is warranted when the incident is too complex for the internal team.

It's also important to have a plan in place for conflict-of-interest escalation. This type of escalation is needed when any members of the support team are suspects. (In the case of First Fidelity, the main suspect was part of the security team. Conflict-of-interest escalation could have alleviated a lot of stress and later personnel problems.)

Keep Good Records

It is wise to develop a reporting mechanism for all break-ins, even those that are resolved without apparent damage to the system. Break-in reports provide an overall picture of the status of network security. Break-in reports can also help pinpoint areas of your network that are vulnerable to security breaches.

Follow Up

After a break-in occurs, you need to assess what happened. Did your staff follow the goals and priorities? What lessons did you learn? What would you do differently next time? Have your systems been restored to a safe state—no back doors?

After any security incident, do the following:

Re-examine Your Policies and Procedures

Thoroughly examine how well your procedures worked and decide whether you need to make changes for the future.

Report the Incident (and How You Handled It) to Management

If you are management, insist that all incidents be reported to you. A standard procedure for reporting any and every break-in provides an overall picture of the status of network security. If reports show that break-ins are chronic or increasing in frequency, it's obvious that security measures need to be updated or augmented. Report protocols can also be used to identify areas of the network that intruders might be targeting for data (e.g., source code for your new chip design).

Have Another Look at That Budget

On paper, everybody loves security. But when it comes to funding, security planning and response are often shorted. “The budget is tight this quarter, so management says incident-response costs will have to wait.” The next thing you know, a year has gone by and the procedures still haven't been written.

The importance of security is easily overlooked. Every once in a while, a major break-in makes some poor soul's company the focal point of *60 Minutes* or CNN. Everybody is suddenly very anxious to set up security measures and make sure that the same thing can't happen to them. Then the spotlight dims, the media go away, and the hacker goes to jail or disappears into cyberspace. Security becomes a nonissue, and management is once again reluctant to include it in the budget.

Marcus Ranum, often referred to as the father of firewalls, once said, “When it comes to security, it usually takes a bullet to the head of the guy standing next to you before management takes notice.” If you are a manager responsible for security, don't take the “bullet” approach to security. The truth is, it costs much more to clean up after a serious break-in than it does to put defenses in place. To minimize those costs in the future, be sure to include requests for adequate funding for security requirements.

Checklist

Use this checklist to determine whether your company is prepared to respond to a break-in. Can you mark a “Yes” beside each item?

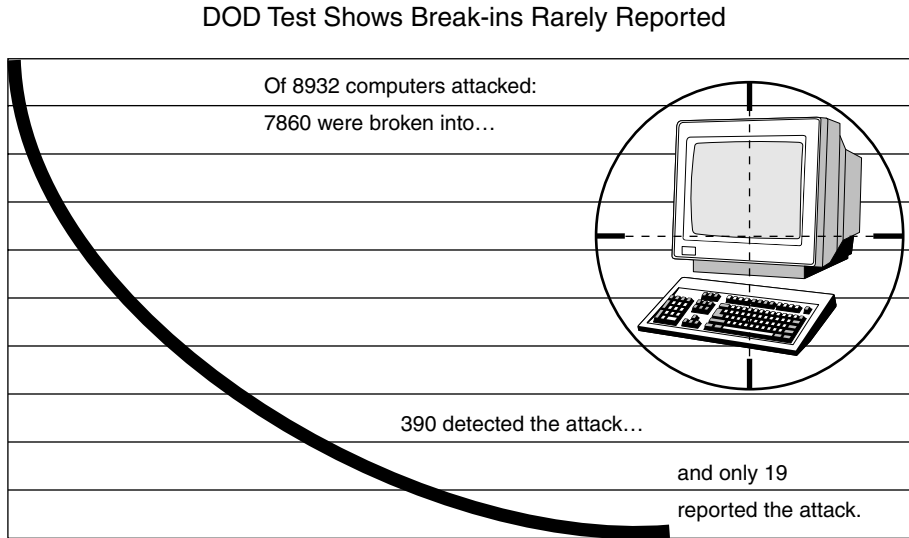
- ___ Do incident-response procedures exist?
- ___ Are procedures understandable and up-to-date?
- ___ Have all key personnel been trained in using the procedures?
- ___ Do the procedures include instructions for contacting a security expert 24-hours-a-day, 7-days-a-week?
- ___ If the security expert does not respond, does a procedure exist for escalating the problem to management?
- ___ Is there a procedure for determining when to contact outside help, and whom to contact?
- ___ Do procedures include notifying the CIO immediately when any break-in occurs, and again when the break-in is resolved?
- ___ Has adequate funding been allotted for developing and maintaining incident responses to break-ins?
- ___ Have key personnel actually attended all required training sessions?
- ___ Have appropriate background checks been conducted on key personnel?
- ___ Are communications between and among the system administration and security groups flowing smoothly?
- ___ Are disaster-recovery plans in place?
- ___ Do all systems have adequate security controls? (“Adequate” here means proven adequate by formal audit results.)
- ___ Are system audit logs enabled?
- ___ Are system logs periodically reviewed?
- ___ Are the tools needed to detect an intrusion installed and operational?
- ___ Can the detection software installed on your network detect unknown attacks?
- ___ Can you detect and prevent attacks on the network and the host (a layered approach to detection)?
- ___ Are attacks easy to trace back on your network?

Final Words

Statistics compiled by CERT show that security violations are more than doubling in number every year. Reported incidents rose from 3,934 in 1998 to 9,859 in 1999, then to 21,1756 in 2000 and 52,658 in 2001. The first quarter of 2002 alone saw another 26,829 reported incidents. Even more frightening, many violations aren’t reported because they are never detected. While 38 percent of CSI’s 2002 survey respondents reported unauthorized use of their Web sites for the previous year, another 21 percent reported that they honestly didn’t know whether or not their sites had been compromised.

With statistics like that, it is easy to see that even if you have no reason to believe that your company has ever experienced a break-in, you may have been the victim of an attack that went unnoticed. In a truly classic study, the Department of Defense (DOD) con-

ducted a test that illustrates just how rarely break-ins are detected and reported (Figure 1-4). This particular test set out to attack 8,932 computers. Of those targeted systems, the attacks succeeded in breaking into 7,860 systems—nearly 88 percent. Yet, only 19 of those attacks were reported—less than .003 percent!



Source: Defense Information Systems Agency

Figure 1-4

Dan Farmer (a well-known computer security researcher) conducted a security survey on high-profile, commerce-oriented World Wide Web Internet sites. The results showed that serious security vulnerabilities exist on the Internet. Out of 1,700 Web servers targeted in this study, over 60 percent of the systems could be broken into or destroyed, and only three sites even noticed the probe.

In the rush to get your systems connected to the Internet, you may also have forgotten about security. Your system may even be in that vulnerable 60 percent. If you're not sure of the current security controls on your Web server (or any other system), conduct a security audit, or call a security expert to evaluate your site for you.

The DOD and Dan's test were both completed several years ago. It is hard to say how many companies today would detect these attacks. Many sites have installed intrusion-detection software and are looking for attacks. If your company has not installed software to detect attacks, it needs to. Don't wait for your company name to be mentioned on CNN.