Laura Chappell

Cyber Crime

It Could Happen to You

Network security has always been important, but it's become especially critical after recent cyber attacks brought some heavy-hitting Internet sites (such as Yahoo.com, CNN.com, Amazon.com, E*Trade.com, and ZDnet.com) to their knees. Are you prepared to handle such an attack on your company's network? Do you know what to watch for? Do you know where the network's weaknesses are? Do you have a comprehensive plan of action in case the network is attacked?

This article identifies some common access points for attacks and describes the types of attacks that brought these Internet sites to their knees. This article also analyzes the typical communications pattern that may precede an attack and shows an actual attack in action. Finally, this article outlines the steps you should take to protect your company's network.

Illustration: Norman Felchle

ATTACK ACCESS POINTS

Knowing a network's weak spots can help you identify possible access points for attack (the areas where hackers enter a network). The following are some common access points for attacks:

- Hosts that are running unnecessary services (such as FTPd)
- Network software that is outdated or unpatched
- Firewalls that are full of holes
- Passwords that are old, obvious, or weak
- Information that is being leaked through services such as telnet, finger, and gopher
- Security that is not well-defined
- Software that is installed on the network without the knowledge of the IS staff

Passive and Active Attacks

There are two primary types of security attacks: passive attacks and active attacks. Passive attacks typically focus on stealing data. For example, a hacker may use a sniffing tool (or even a protocol analyzer) to read passwords, usernames, e-mail messages, and even the data that crosses the wire.

Note. In this article, the term *sniffer* is used to identify programs that can capture data crossing the wire. Network Associates also has a network analyzer called *Sniffer*, which is identified in this article with a capital "S."

When I do onsite visits, I typically start by capturing approximately 50 MB of data (with no filters applied). In the evening after the onsite visit, I peruse the trace file—reading the hex portion of as many packets as possible. In the hex portion of packets, I have found passwords, usernames, e-mail messages, and, in one case, financial data (payroll tax information). You should capture and analyze some data on your company's network to see what is being sent across that network. Remember, anyone can place a sniffer on your company's network.

Active attacks, on the other hand, attempt to cause harm typically through system faults or brute force. Most active attacks attempt to overload the victim's computer to the point that it either slows to an unusable crawl, hangs, or completely crashes. The Distributed Denial of Service attack that occurred during the week of February 13, 2000, is a perfect example of an active attack. Hundreds of computers overloaded several selected victims including Yahoo.com, E*Trade.com, and CNN.com.

DENIAL OF SERVICE ATTACKS

Recently, the most popular attacks have been Denial of Service and Distributed Denial of Service attacks. In a typical Denial of Service attack, a single host attempts to overwhelm a victim by saturating its processor with repeated service requests.

More deadly than Denial of Service attacks, Distributed Denial of Service attacks can use an entire army of computers to attack the victim. In most cases, these computers do not even know they have been enlisted in this attack. The attack process itself is a standard Denial of Service attack.

The following are common types of Denial of Service attacks:

- **User Datagram Protocol (UDP) or Internet Control Message Protocol (ICMP) Flood.** This brute force attack sends a continuous stream of UDP echo request packets (port 7) or ICMP echo request packets (pings) to a host. These attacks may have the address of another "innocent" device in the source field, victimizing a third-party. UDP-based floods may also use the chargen port (character generator, UDP port 19). You should disable the echo and chargen UDP ports unless these ports are absolutely necessary.
- **SYN Flood.** Typically, TCP connections require a three-way handshake. (For more information about the TCP handshake, see "Inside the TCP Handshake," *NetWare Connection*, Mar. 2000, pp. 34–36. You can download this article from

http://www.nwconnection.com/past.) In a SYN flood, the attack system transmits a stream of unique TCP SYN packets (the first packet of the TCP handshake sequence). When the attack system receives the ACK SYN response packets from the victim, however, the attack system does not transmit ACK packets to finish the connection. As a result, the victim creates numerous half-opened TCP connections and must wait for each connection to timeout before freeing up the connection resources.

You can use the NETSTAT utility to identify connection states. Too many connections in the SYN_ RECEIVED state indicate a possible attack is underway.

- **Smurf Attack.** This attack is similar to the UDP or ICMP flood except that multiple devices are pinged with the victim's source address. The "innocent" devices send replies to the victim's source address, thereby overloading the victim. The Smurf attack is difficult to detect, especially when the attacker uses a false source IP address—a trick called *IP spoofing*.
- **Land Attack.** In this attack, a TCP SYN packet uses the victim's IP address in both the source and the destination address fields in the IP header. A land attack can cause the victim to lock up or to respond slowly. To fix this problem, many vendors have patched their TCP/IP stack to discard packets that have identical source and destination IP addresses.

There are, of course, hundreds of variations on these attacks. For example, two of the most popular Distributed Denial of Service attacks are Trinoo and Tribe Flood Network (TFN).

TRINOO ATTACK

Trinoo is a Distributed Denial of Service program that uses a UDP flood to disable the victim. A trinoo network consists of an attacker system, several compromised systems, including one or more masters (referred to as *handlers*) and one or more daemon systems (referred to as *agents*), and one or more victims.

The Trinoo attack is a three-step process, as shown in Figure 1:

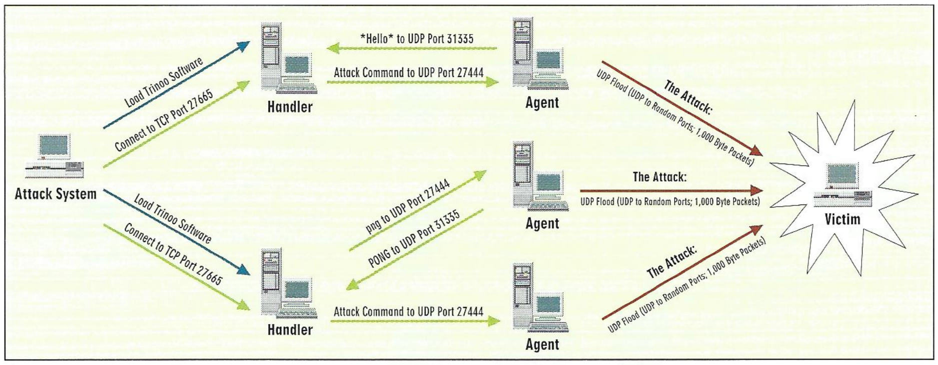

Attack System
Load Trinoo Software
Connect to TCP Port 27665
Handler
Hello to UDP Port 31335
Attack Command to UDP Port 27444
Agent
The Attack:
UDP Flood (UDP to Random Ports; 1,000 Byte Packets)
Load Trinoo Software
Connect to TCP Port 27665
png to UDP Port 27444
PONG to UDP Port 31335
Agent
The Attack:
UDP Flood (UDP to Random Ports; 1,000 Byte Packets)
Victim
Handler
Attack Command to UDP Port 27444
Agent
The Attack:
UDP Flood (UDP to Random Ports; 1,000 Byte Packets)

Figure 1. The Trinoo architecture

Step 1. To begin building the Trinoo network, the attack system loads the Trinoo program on one or more compromised computers that become the handlers and agents. (To find out how Distributed Denial of Service programs can be installed on systems, visit http://www.washington.edu/People/dad.)

Note. The terms *handler* and *agent* are in compliance with the terminology defined at the Distributed-Systems Intruder Tools Workshop (Nov. 2–4, 1999). For more information, visit http://www.cert.org/reports/dsit_workshop.pdf.

Step 2. The agents send a UDP packet that contains the text string "*HELLO*" to the handler (using UDP port 31335) to let the handler know the agent daemons are loaded and ready. When the attack system sends the attack command (using destination TCP port 27665), the handler sends a message to the agents (using UDP port 27444) to launch the attack.

The handler can check to see which agents are running by sending a text message ("png") to UDP port 27444. All agents with active daemon systems will respond with the text string "PONG." In this way, the handler can keep a current list of all agents in the Trinoo network.

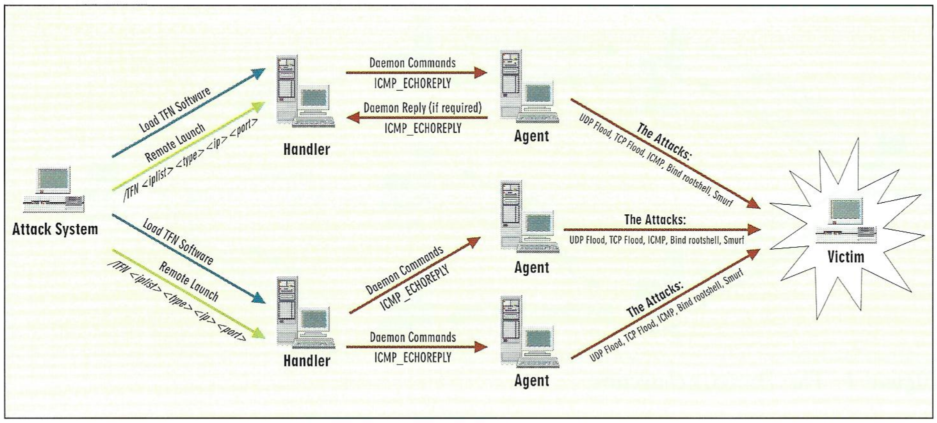


***Figure 2.** The TFN architecture*

Step 3. After receiving the command to launch an attack, the agent sends a UDP flood to random port numbers on the victim. The default packet size is 1,000 bytes (although this packet size can be changed in the program).

Catching a Trinoo Attack

As Figure 1 shows, the Trinoo attack uses specific port numbers. (See p. 37.) If you configure a network analyzer to filter on all traffic sent to the TCP port 27665, UDP port 31335, UDP port 27444 (and now UDP port 34555 for Windows systems), you may be able to catch a Trinoo attack in action.

Of course, Trinoo developers are probably building more robust versions of this attack. To find out more information about the "fingerprints" of the latest Trinoo attacks, visit the Computer Emergency Response Team (CERT) web site (http://www.cert.org. To read the CERT Incident Notes for Trinoo, visit http://www.cert.org/incident_notes/IN-99-07.html. As this article goes to press, an agent scanner for this attack is in beta testing. For more information, visit Dave Dittrich's web site at http://www.washington.edu/People/dad.

TFN ATTACK

The TFN attack is more complex than the Trinoo attack because the TFN attack can execute several attack sequences. The TFN architecture is also slightly different than the Trinoo architecture. (See Figure 2.) In addition, TFN uses a variation of the PING function to send commands between the handler and the agents.

Like the Trinoo attack, the TFN attack is a three-step process:

Step 1. The TFN attacker must load the TFN software onto compromised computers. For example, in the Distributed Denial of Service attack that occurred over the Internet in February, the attacker used computers at Stanford University and several other universities.

Step 2. To launch the attack, the attack system simply needs remote access to the handler. The attack system can then send the following command to launch the executable:

```
/TFN <iplist> <type> <ip> <port>
```

The variables are replaced with the following information:

<iplist> is replaced with a list of numerical hosts that are ready to flood (the daemon systems).

<type> is replaced with one of the following variables, which determines the type of TFN attack:

- 1 for a UDP echo flood
- 2 for a TCP flood
- 3 for an ICMP (ping) flood
- 4 to obtain root access
- 5 for a Smurf attack (to target, then to broadcast)

<ip> is replaced with the target IP addresses, separated by @ if more than one target.

<port> is replaced with the port number (such as 0=random). This is required for a SYN flood.

Step 3. The handlers use ICMP_ECHOREPLY messages to notify the agents to begin the attack.

Catching a TFN Attack

As Figure 2 shows, the TFN attack uses a strange variation of the PING command. Typically, a PING packet has two parts:

- ICMP_ECHOREQUEST
- ICMP_ECHOREPLY

In the case of a TFN attack, only the ICMP_ECHOREPLY is used. Most likely this ensures that uninvolved systems do not respond to the commands sent by the TFN client. Unfortunately, most network analyzers do not have a filter for unsolicited responses: You can only configure network analyzers to view all ICMP echo requests and replies. To detect a TFN attack, you will need to search for an echo reply packet that is not preceded by an echo request packet.

To protect your company's network against a TFN attack, a dedicated security program will be required. As this article goes to press, an agent scanner for this attack is in beta testing. For more information, visit Dave Dittrich's web site at http://www.washington.edu/People/dad.

Enquiring Minds Want to Know About Cyber Crime

For more information about recognizing potential attacks and protecting your company's network from hackers, visit the following web sites:

- **http://www.cert.org.** Computer Emergency Response Team (Carnegie Mellon University)
- **http://www.auscert.org.au.** Australian Computer Emergency Response Team
- **http://www.htcia.org.** High Technology Crime Investigation Association
- **http://www.netanalysis.org.** Trace files of attacks and typical communications
- **http://www.nai.com/nai_labs/asp_set/cybercop.asp.** CyberCop Research Center (Network Associates)
- **http://www.sans.org.** System Administration, Networking and Security Institute
- **http://www.washington.edu/People/dad.** Dave Dittrich's fantastic hacking resources
- **http://www.novell.com/corp/security.** Novell's security web site

STACHELDRAHT ATTACK

Another attack that has received a fair amount of press is the Stacheldraht attack. (Stacheldraht is a German word meaning "barbed wire.") Stacheldraht is a combination of Trinoo and TFN. However, Stacheldraht relies on TCP, rather than UDP, for transport. Stacheldraht uses TCP port 16660 between the attack system and the handler and TCP port 65000 between the handler and the agent.

Stacheldraht can be automatically upgraded through Remote Procedure Calls (RPCs). Upon demand, all agents must delete their current image and download a new image from another site. Most likely, the updated file has been placed on a compromised system without the administrator's knowledge.

Stacheldraht handlers and agents periodically exchange ICMP_REPLY packets. They use ICMP ID field value 666 for the handler and ICMP ID field value 667 for the agent. You should be able to capture this type of traffic with a well-written filter for network analyzers.

To detect such Stacheldraht attacks, you can download a program that Dave Dittrich, Marcus Ranum, and others have written. (To download this program, visit http://www.washington.edu/People/dad or http://packetstorm.securify.com/distributed/sickenscan.tar.) You can also run the NETSTAT utility on a host to display its active ports.

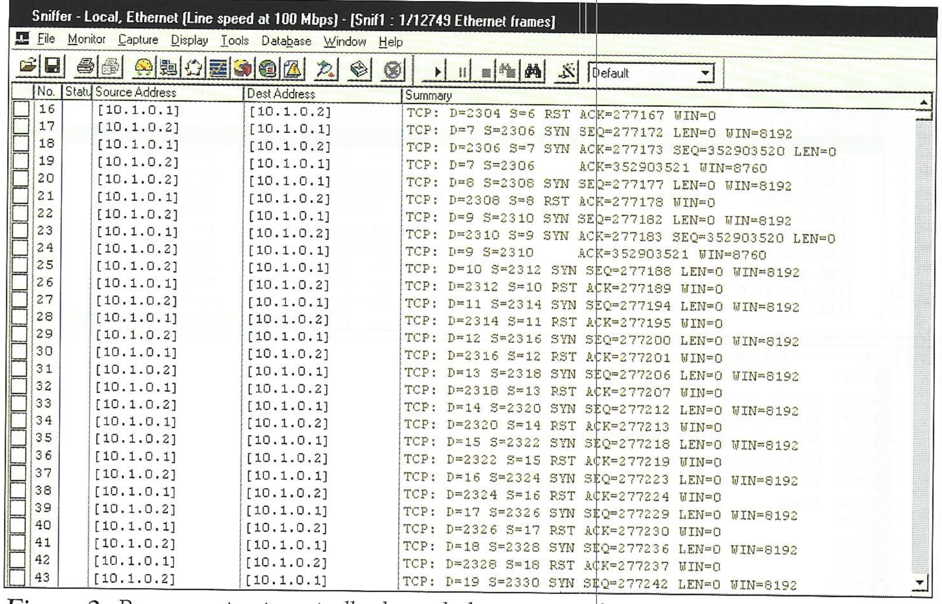
IDENTIFYING "SUSPICIOUS BEHAVIOR"

Having the right set of security tools will help you identify specific communications patterns that could be considered suspicious. For example, in Figure 3, host 10.1.0.2 appears to be performing a port scan on host 10.1.0.1. Port scanning is used to determine what daemons or processes are running on a device. The device performing the scan sends a series of packets with different destination port numbers. Based on the replies received, this device can build a list of active ports.

A hacker uses port scanning to find out which ports are active. Typically, a hacker will run a port scan over both UDP and TCP.

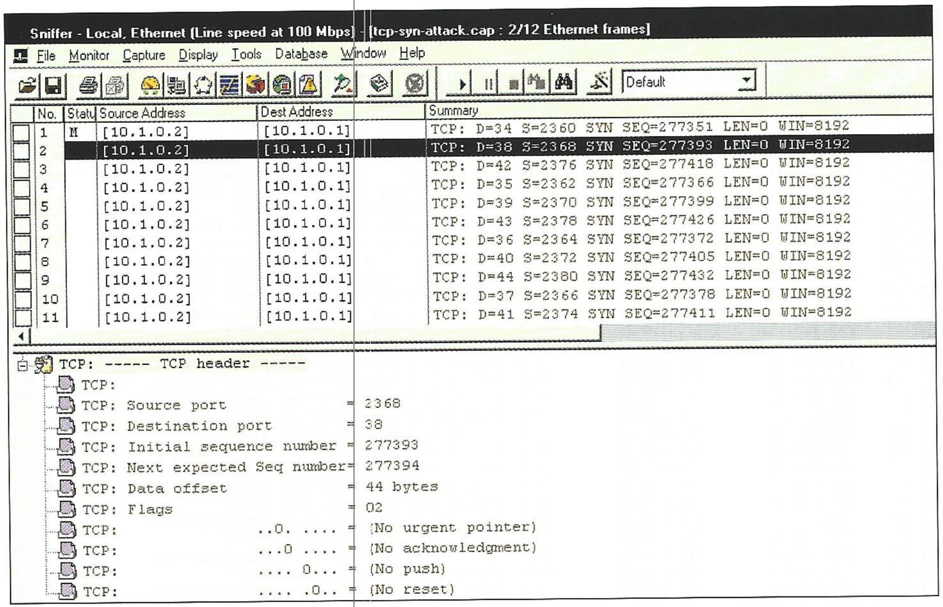
Figure 3 depicts an unsophisticated TCP port scanning operation—the ports are being scanned in succession. Typically a randomizing feature is used to camouflage port scanning activity. The device (10.1.0.2) performing the scan in Figure 3 has found two services that are active on the target (10.1.0.1): port 7 Echo and port 9 Discard.

Now that the scanner and, therefore, the hacker know the ports that are active, the hacker can begin to exploit these services and look for weaknesses. For example, the echo process defines that every character sent to the target will be echoed back. An attack uses echo and chargen to echo back data.


Sniffer - Local, Ethernet (Line speed at 100 Mbps) - [Snif1 : 1/12749 Ethernet frames]
File
Monitor
Capture
Display
Tools
Database
Window
Help
Default

```
No. Statu Source Address  Dest Address   Summary
16        [10.1.0.1]      [10.1.0.2]     TCP: D=2304 S=6 RST ACK=277167 WIN=0
17        [10.1.0.2]      [10.1.0.1]     TCP: D=7 S=2306 SYN SEQ=277172 LEN=0 WIN=8192
18        [10.1.0.1]      [10.1.0.2]     TCP: D=2306 S=7 SYN ACK=277173 SEQ=352903520 LEN=0
19        [10.1.0.2]      [10.1.0.1]     TCP: D=7 S=2306     ACK=352903521 WIN=8760
20        [10.1.0.2]      [10.1.0.1]     TCP: D=8 S=2308 SYN SEQ=277177 LEN=0 WIN=8192
21        [10.1.0.1]      [10.1.0.2]     TCP: D=2308 S=8 RST ACK=277178 WIN=0
22        [10.1.0.2]      [10.1.0.1]     TCP: D=9 S=2310 SYN SEQ=277182 LEN=0 WIN=8192
23        [10.1.0.1]      [10.1.0.2]     TCP: D=2310 S=9 SYN ACK=277183 SEQ=352903520 LEN=0
24        [10.1.0.2]      [10.1.0.1]     TCP: D=9 S=2310     ACK=352903521 WIN=8760
25        [10.1.0.2]      [10.1.0.1]     TCP: D=10 S=2312 SYN SEQ=277188 LEN=0 WIN=8192
26        [10.1.0.1]      [10.1.0.2]     TCP: D=2312 S=10 RST ACK=277189 WIN=0
27        [10.1.0.2]      [10.1.0.1]     TCP: D=11 S=2314 SYN SEQ=277194 LEN=0 WIN=8192
28        [10.1.0.1]      [10.1.0.2]     TCP: D=2314 S=11 RST ACK=277195 WIN=0
29        [10.1.0.2]      [10.1.0.1]     TCP: D=12 S=2316 SYN SEQ=277200 LEN=0 WIN=8192
30        [10.1.0.1]      [10.1.0.2]     TCP: D=2316 S=12 RST ACK=277201 WIN=0
31        [10.1.0.2]      [10.1.0.1]     TCP: D=13 S=2318 SYN SEQ=277206 LEN=0 WIN=8192
32        [10.1.0.1]      [10.1.0.2]     TCP: D=2318 S=13 RST ACK=277207 WIN=0
33        [10.1.0.2]      [10.1.0.1]     TCP: D=14 S=2320 SYN SEQ=277212 LEN=0 WIN=8192
34        [10.1.0.1]      [10.1.0.2]     TCP: D=2320 S=14 RST ACK=277213 WIN=0
35        [10.1.0.2]      [10.1.0.1]     TCP: D=15 S=2322 SYN SEQ=277218 LEN=0 WIN=8192
36        [10.1.0.1]      [10.1.0.2]     TCP: D=2322 S=15 RST ACK=277219 WIN=0
37        [10.1.0.2]      [10.1.0.1]     TCP: D=16 S=2324 SYN SEQ=277223 LEN=0 WIN=8192
38        [10.1.0.1]      [10.1.0.2]     TCP: D=2324 S=16 RST ACK=277224 WIN=0
39        [10.1.0.2]      [10.1.0.1]     TCP: D=17 S=2326 SYN SEQ=277229 LEN=0 WIN=8192
40        [10.1.0.1]      [10.1.0.2]     TCP: D=2326 S=17 RST ACK=277230 WIN=0
41        [10.1.0.2]      [10.1.0.1]     TCP: D=18 S=2328 SYN SEQ=277236 LEN=0 WIN=8192
42        [10.1.0.1]      [10.1.0.2]     TCP: D=2328 S=18 RST ACK=277237 WIN=0
43        [10.1.0.2]      [10.1.0.1]     TCP: D=19 S=2330 SYN SEQ=277242 LEN=0 WIN=8192
```

Figure 3. *Port scanning is typically the prelude to an attack.*


Sniffer - Local, Ethernet (Line speed at 100 Mbps) - [tcp-syn-attack.cap : 2/12 Ethernet frames]
File Monitor Capture Display Tools Database Window Help
Default

```
No. Statu Source Address  Dest Address   Summary
1   M     [10.1.0.2]      [10.1.0.1]     TCP: D=34 S=2360 SYN SEQ=277351 LEN=0 WIN=8192
2         [10.1.0.2]      [10.1.0.1]     TCP: D=38 S=2368 SYN SEQ=277393 LEN=0 WIN=8192
3         [10.1.0.2]      [10.1.0.1]     TCP: D=42 S=2376 SYN SEQ=277418 LEN=0 WIN=8192
4         [10.1.0.2]      [10.1.0.1]     TCP: D=35 S=2362 SYN SEQ=277366 LEN=0 WIN=8192
5         [10.1.0.2]      [10.1.0.1]     TCP: D=39 S=2370 SYN SEQ=277399 LEN=0 WIN=8192
6         [10.1.0.2]      [10.1.0.1]     TCP: D=43 S=2378 SYN SEQ=277426 LEN=0 WIN=8192
7         [10.1.0.2]      [10.1.0.1]     TCP: D=36 S=2364 SYN SEQ=277372 LEN=0 WIN=8192
8         [10.1.0.2]      [10.1.0.1]     TCP: D=40 S=2372 SYN SEQ=277405 LEN=0 WIN=8192
9         [10.1.0.2]      [10.1.0.1]     TCP: D=44 S=2380 SYN SEQ=277432 LEN=0 WIN=8192
10        [10.1.0.2]      [10.1.0.1]     TCP: D=37 S=2366 SYN SEQ=277378 LEN=0 WIN=8192
11        [10.1.0.2]      [10.1.0.1]     TCP: D=41 S=2374 SYN SEQ=277411 LEN=0 WIN=8192

TCP: ----- TCP header -----
   TCP:
   TCP: Source port               = 2368
   TCP: Destination port          = 38
   TCP: Initial sequence number   = 277393
   TCP: Next expected Seq number= 277394
   TCP: Data offset               = 44 bytes
   TCP: Flags                     = 02
   TCP:                 ..0. .... = (No urgent pointer)
   TCP:                 ...0 .... = (No acknowledgment)
   TCP:                 .... 0... = (No push)
   TCP:                 .... .0.. = (No reset)
```

Figure 4. *The classic TCP-SYN attack*

This attack is called *ping pong* and is quite simple to execute. A hacker simply port scans the network to find a computer that supports echo and a computer that supports chargen. The hacker builds a packet with a data sting inside it and addresses this packet from one computer's echo port to the other computer's chargen port. The effect is a ping-pong communication back and forth between the two victim computers. I strongly recommend that you disable echo and chargen processes on your company's network. (To review the UDP and TCP echo process, see the traces in the udp-echo.cap/udp-echo.prn and tcp-echo.cap/tcp-echo.prn files at http://www.netanalysis.org.)

Figure 4 shows another hacking attempt—a TCP SYN attack. The hacker is sending a blast of TCP connection request packets. Because the hacker has used a unique source port number in each request, each request requires a separate connection. (To view the entire trace file, see the tcp-syn-attack.cap/tcp-syn-attack.prn file at http://www.netanalysis.org.)

PREPARATION AND PREVENTION

You must be prepared to detect network intrusions. Preparation includes defining a security policy, implementing security procedures, and developing or activating the tools required to detect these intrusions. For example, CERT recommends the following four-step plan:

1. Establish a security policy and procedures that prepare your company to detect signs of intrusion.
2. Identify and enable system and network logging mechanisms.
3. Identify and install tools that help detect signs of intrusion.
4. Generate information required to verify the integrity of your company's systems and data.

CERT maintains a library of material that focuses on building and implementing a security policy. I strongly recommend that you spend an hour or two perusing the CERT web site (http://www.cert.org). I also recommend that you find out more about cyber attacks. "Enquiring Minds Want to Know About Cyber Crime" provides some good resources on where to start. (See p. 38.)

The tools that you implement on your network should be able to help you detect the following events:

- Password cracking
- Execution of unauthorized programs
- Installation of tools (such as network analyzers) that may be used to break into other systems
- Internet Relay Chat (IRC)
- Intruder use of unexpected or unrecognized hosts
- Intruder access during non-business hours
- Intruder file transfers of tools to be used in launching subsequent attacks (such as TFN)
- Virus infiltration
- System or key image file changes

Most vendors also maintain a security advisory system to announce possible security holes and the patches to fix such violations. Keeping software up-to-date is one of the most effective methods of securing your company's network.

This article is really only the tip of the cyber crime iceberg (remember the Titanic?). As the Internet, intranets, extranets, and private networks continue to grow, so does the chance of being hacked. February's Distributed Denial of Service attack was a wake-up call to everyone. It's time to get network security in order—now that all the year-2000 fuss has quieted down, consider setting up that year-2000 command center as a Cyber Security Center. Your company may not survive without it.

Laura Chappell is the senior protocol analyst for NetAnalysis Institute. She writes books on packet-level communications, including "Introduction to Network Analysis" and "TCP/IP Analysis and Troubleshooting," available online at http://www.podbooks.com and at Amazon Books. ●