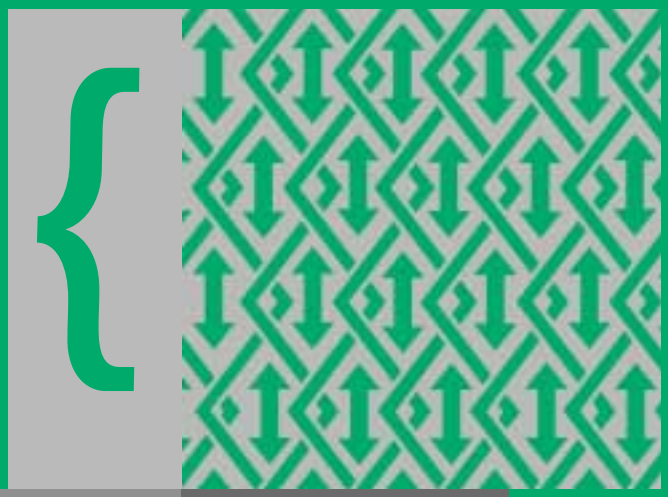
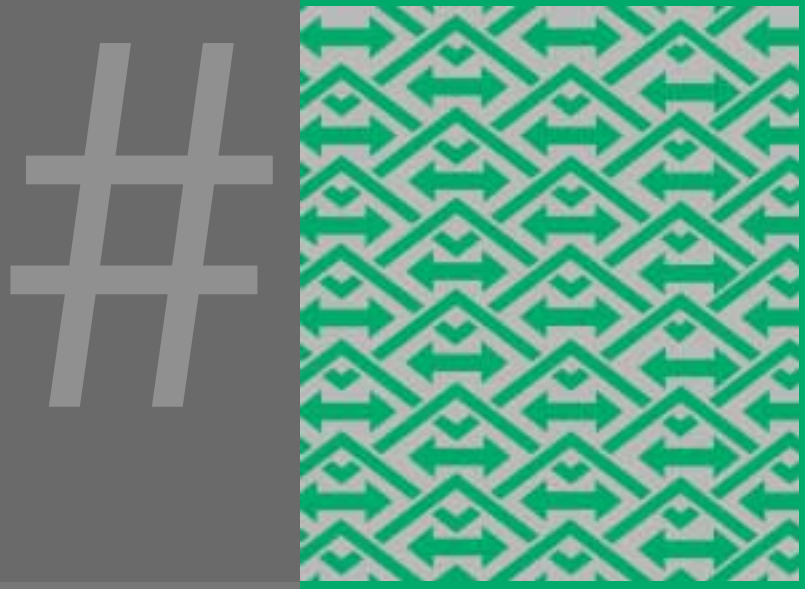
;login:

THE MAGAZINE OF USENIX & SAGE
October 2000 • volume 25 • number 6



inside:



SECURITY
Use Your Local Tools

USENIX & SAGE
The Advanced Computing Systems Association &
The System Administrators Guild

use your local tools

There are some times when a sysadmin has to make do with what is available. This is a story about one of those times.

by Barbara Dijker

Barbara Dijker is currently SAGE president. She's been sysadminning for about 12 years and runs a couple of ISPs.

<barb@usenix.org>

I received an email message from someone in the UK with the subject "IMPORTANT: do not ignore." The lengthy content went on and on about how they suspected that a host on my network was running a distributed-denial-of-service (DDoS) agent called Stacheldraht. Other than that, the message was strangely uninformative. It contained a link to a site with an analysis of various DDoS systems and tools to find them. But it provided no evidence of how they came to the conclusion my network was involved. Further, it said not to ask for evidence because it would be of no use.

Hmm. Was this a joke? I have to admit I haven't been keeping up with the latest script-kiddie toys. So I wasn't sure if I could have Stacheldraht in my midst and not know it. I decided rather than ignore the message, I'd at least find out if it could be true. I read up on DDoS. Stacheldraht means "barbed wire" in German. I'll let you read the full analysis at your leisure some other time at *<http://staff.washington.edu/dittrich/misc/ddos/>*. But here is the brief overview of what I learned and how I managed to find the culprit in less than ideal circumstances. I did ask the person who contacted me to at least send me the time stamps on his log entries that implicated my site. I figured I could use those to narrow down the part of the haystack I needed to look in.

Stacheldraht is based on Tribe Flood Network and incorporates features of trinoo. Agents are installed on as many hosts as possible – on hosts presumably targeted by automated probes and compromised by standard root kits. Linux and Solaris hosts tend to be the favorite targets. The agents are monitored and controlled by "handlers." In order for this to happen, the agents communicate with the handlers using ICMP echo-reply packets. Yes, just reply, no echo request. This is probably because a firewall is more likely to pass echo-reply than echo-request packets. The data portion of the ICMP packet is used to transmit control and status information. But it is encrypted. The encryption is easy to break if the agent was installed with the stock key (provided on the analysis Web site). Once installed, a Stacheldraht agent can be commanded to launch one of several popular DoS attacks: ICMP flood, SYN flood, etc. The analysis Web site also provided text strings commonly found in the program like "sickem." Stacheldraht also spoofs the source IP address of the packets it sends during an attack. If the local router will let it, it will spoof the entire IP address. Otherwise it will spoof only the last byte. It verifies this by sending a spoofed packet to its handler (with the real source address in the payload) and waiting for a confirmation.

The attack packets received by the site in the UK apparently showed my network as the source IP, but with random fourth bytes. That's why they said the log entries would be of no use. But the packets could have also listed my network while being sent from elsewhere. While that was a possibility, I figured it was still worth looking for unusual activity.

The network in question here is an ISP network. It is a /24, or "class C" network. There are over 100 hosts on that network. Each is owned and administered by a different company and person with few exceptions. I don't have user or root access on any hosts on that network except my monitoring station and my sniffer. So logging in to each system and looking for signs of compromise was not an option. That would be tedious and inefficient anyway. The monitoring I do is purely for billing. There is no firewall.

The first thing I did was check my traffic logs – one for (almost) each host. The date of the incident was a few weeks prior, so the MRTG data was reduced enough that it was

As you might expect with a large (switched) network of strange machines, bizarre things happen frequently.

difficult to see any anomalies. I found four candidate suspects with what appeared to be about 100kbps more outgoing traffic than usual. We notified their system administrator to look for possible compromise.

That produced nothing of course. They each said their system was impenetrable. They probably didn't even look. So I went back to the analysis site to look for tools. There are a few promising ones. I tried to install the two that were recommended. The problem I ran into is that they all eventually require libpcap to build custom packet headers. I don't own a Linux box. I tried a simple make where appropriate on BSDI, FreeBSD, and Solaris. All failed! I wasn't in the mood to port someone else's code. Nessus looked intriguing. I decided that would be best installed when I had the time to learn it properly – to decide an optimal and long-term configuration. So I opted to use a tool already on hand with which I'm comfortable and familiar: my sniffer.

As you might expect with a large (switched) network of strange machines, bizarre things happen frequently. My sniffer is one of my best friends. The average traffic level on this network is in excess of 5Mbps. Long ago I invested in a sniffing tool that was flexible and useful and that provides quick results. It's a dedicated beefy and zippy PC with a high-quality fast Ethernet card. I use software called Observer by Network Instruments. Alas, it has to run on NT. But it met my criteria.

The first thing I asked my sniffer to do was track and capture all ICMP echo-reply packets. In just a few minutes, I had a list sorted by number of packets. The top hosts on that list became suspect. I browsed the packet-header dumps of some of the individual packets sent and received by the top suspect. Indeed they looked like they could be Stacheldraht because of size and ICMP ID. I didn't go to the trouble of decrypting the payload to see for certain. However, I was amazed at how many of the hosts on our network send (and receive) many and frequent pings as a matter of course. That made it more difficult to use this as definitive evidence.

In the meantime, we started to suffer some minor issues on our router – it was noticeably slower than it should have been. A quick look there indicated that the packets it was routing per second increased 50-fold from typical! The problem then went away as soon as we found it, of course. So the next thing we asked the sniffer to do was to contact us when the packets per second went above a threshold. Sure enough, within a day we had a trigger. The top sender of tons of tiny packets was indeed the same host that was the champion of ICMP echo replies.

At that point we captured some of the packets this host was sending, and bingo. All the packets were TCP ACKs. They were sent repeatedly and in large quantity to a short list (<10) of destinations. Interestingly the quantity was not large enough to look unusual in our MRTG traffic graphs – because they graph bits per second, not packets per second. The source IP address used indeed had a spoofed fourth byte. We know it was spoofed because the Ethernet address was the same. (That's what we used to filter the packet capture.) We have quite a few hosts that are assigned many IP addresses for virtual Web hosts. But typically you'll only see those as the IP destination, while the source IP used is the primary IP of the host. We confirmed that the IP addresses being used to source these packets were assigned to different hosts entirely.

OK. We caught one. It was a Linux box. Then we needed to make sure our Ethernet-address database wasn't old or wrong. We unplugged its network cable to verify the packets stopped. Then we plugged it in again and verified the packets reappeared. We were confident we had the right host. Most sniffers will learn and build your Ethernet-

to-IP address table for you based on the source addresses of normal packets. You just need to make sure you do that when no one is spoofing.

We left the system unplugged and contacted the owner/administrator. They were in a meeting all day and couldn't be bothered. So we asked for root access. Our goal was to clean up the system enough so that we could bring the system back online to serve mail and Web.

We indeed found Stacheldraht. It was called /bin/in.sysched. We also found that the hacker installed their own ssh in /usr/sbin/in.amdq with its configuration files in /dev/sdc0. They installed their own "pam" authentication module in /dev/sde0. In addition, they installed their own bind (resolve.conf was still configured as a client) and their own wu-ftpd – both of which were running and presumably had back doors. Of course they replaced /bin/login and /bin/ps. Finally, some files had been made immutable so that even root could not set them back without additional effort. We instructed the customer to "nuke it from orbit" – a fresh installation was recommended.

The customer sent someone out, and they spent about five hours reinstalling the operating system from scratch, upgrading to the most recent version of their brand of Linux in the process. Within five days, they were hacked and running Stacheldraht again. The customer had rebuilt the system from scratch, but they neglected to install known patches. They thought that the latest version must be secure. Further, they never bothered to check their mail server or Web server for vulnerabilities. A quick check showed both were vulnerable.

The moral to the story? If you monitor traffic, monitor bps and pps. Have the tools you need. Use the tools you have. Know how to use them properly.

If you monitor traffic, monitor bps and pps.