SECURITY PATROL
by Vincent LeVeque

Telnet Tells Too Much

QUESTION: Our network administrator used a "sniffer" program to demonstrate that AS/400 passwords could be revealed by listening over the network. We found, however, that using Telnet 5250 (TN5250) did not reveal the password. Does TN5250 use some sort of encryption that ordinary Telnet doesn't?

ANSWER: First, I should explain exactly what the function is of a so-called "sniffer" program, which is named after the original sniffer program, the **Network General** Sniffer. Network-traffic sniffers (or snoopers as I prefer to call them) eavesdrop on all network traffic on a particular subnet. The snooper software places the PC's Ethernet card into promiscuous mode, whereby it retrieves all traffic passing by, whether or not it is addressed to that particular machine. The traffic is then formatted into ASCII to make it more human-readable. A network eavesdropper can read passwords or other sensitive information passing from system to system, with neither the source nor the target system any the wiser. The information observed by a snooper is determined by the method the workstation uses to communicate with the host. In your case, you have a couple of alternatives: Telnet and TN5250.

Telnet is a longtime Internet standard for terminal emulation. It provides host-to-terminal communication similar to that of an old-style dumb ASCII terminal, such as a **Digital** VT100. Telnet sends each character entered at the keyboard to the host computer in plain, easily understood ASCII code. To a network snooper, it is thus easy to eavesdrop on traditional Telnet traffic.

The other standard used for AS/400 terminal communications is TN5250. The TN5250 data stream is a standard for terminal and printer communications with AS/400 systems over TCP/IP networks. It is described in the Internet Engineering Task Force (IETF) document RFC 1205 (*www.ietf.org/rfc/rfc1205.txt?number=1205*) , which is supported by related **IBM** documentation, notably *Functions Reference.* Availability of this documentation has enabled third-party vendors to provide some fairly decent TN5250 products; a freeware version of TN5250 even exists for Linux.

Unfortunately, TN5250 does not encrypt the data stream between the terminal and the AS/400. You were unable to view the password because the 5250 data stream is more complex than standard ASCII-based Telnet. To illustrate, I have replicated your network administrator's experiment. Figure 1 shows the results of a common snooping tool called sniffit; user ID and password are shown clearly. Figure 2 shows the same logon procedure using the freeware version of **MochaSoft** (*www.mochasoft.dk*), a common third-party TN5250 application. No obvious user information is displayed; the 5250 data stream obscures information in two ways. First, included with the information itself is a variety of "metainformation" describing screen formatting, display attributes, field attributes (e.g., underlining, column separators, etc.), screen operations, and all the various other items you could code in a DDS display file. Second, buried within all this metainformation is the clearly visible screen data (clearly visible if you can read EBCDIC).

The good news here is that standard, off-the-shelf, network-snooping software can't easily grab AS/400 passwords off the wire. Off-the-shelf snooping software is unaware of the metainformation embedded in the 5250 data stream and, moreover, is ignorant of EBCDIC. Your typical network hacker is also most likely unfamiliar with the EBCDIC character set and 5250 data stream format
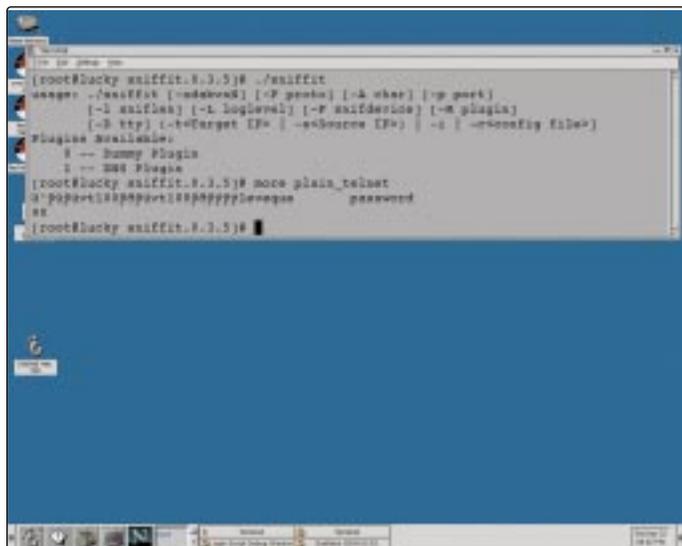

Figure 1: Nothing up the sleeve: User ID and password are shown clearly with the sniffit tool.
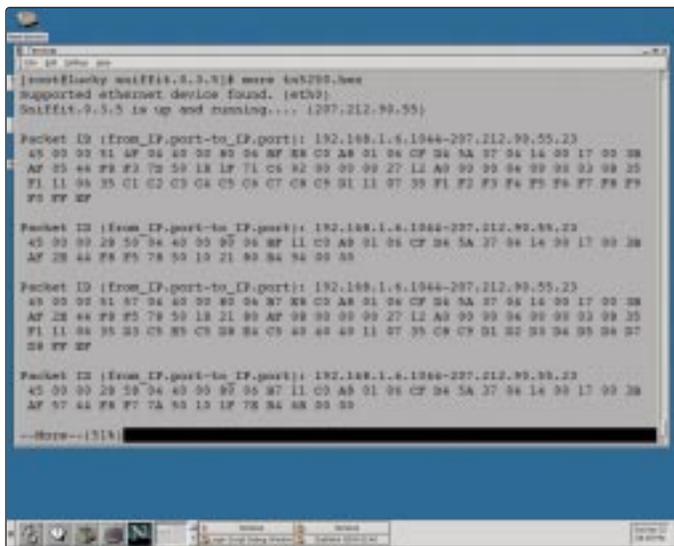

Figure 2: The 5250 data stream throws off typical hackers by camouflaging security information in a bunch of "metainformation."

and is hence unable to make sense of Figure 2. The bad news is that someone who is knowledgeable about TN5250 could probably figure it out. The 5250 data stream format used by TN5250 is documented, and, even if it weren't, it would take just a little experimentation to figure out where the password is hidden in Figure 2. Obscuring data is not the same as encrypting it. Encryption creates true security in that, even if you know the encryption method and general format of the result, you still cannot decipher the message hidden within it.

Grabbing passwords off a local network is a proven technique for breaking into systems. When my company does security assessments, I find that listening to a network hub often provides user IDs and passwords quickly. To guard against password sniffing, you have several options:

- *Use switched hubs.* This limits both the range over which information is broadcast and the opportunities for snooping around any given sign-on process.
- *Use encrypted protocols for user access.* As of V4R4, the AS/400 provides Secure Sockets Layer (SSL) with clients for FTP and Telnet. SSL negotiates an encrypted session before you are even provided a sign-on screen. Client Access Telnet clients support SSL, as do some third-party TN5250 products. Standard Web browsers also support SSL, so applications that require sign-on via a Web-enabled form should establish an SSL session before prompting the user to sign on. Just make sure that you're both using the same level of SSL, which comes in several flavors.
- *Consider using Client Access for online AS/400 network access.* Client Access can provide password security by authenticating use of a cryptographic token rather than by sending a clear-text password over networks.
- *Establish a clear policy restricting use of network-snooping software to authorized network administrators for legitimate diagnostic purposes.* Ensure that disciplinary action is taken for those who violate this policy. To assist in finding unauthorized software on desktop machines, combine this with management of desktop system software via **Microsoft**'s Systems Management Software (Microsoft's platform for network management) or other vendors' products.
- *Consider using sniffer detection tools.* Tools such as **L0pht**'s "antisniffer" software (*www.l0pht.com*) note anomalies in how snoopers respond to network traffic, and they provide leads as to the possible existence of snoopers on a network. These tools are new and their effectiveness has yet to be proven in widespread use, so use some caution. (By the way, *L0pht* is spelled with a

zero, not with the letter *O.*)

Lastly, it should be noted that the TN5250 standard is constantly being revised and updated. The IETF has published a draft of proposed TN5250 enhancements (RFC 1205), among which are included provisions for a strong, cryptographically based authentication method similar to that used for Advanced Program-to-Program Communications (APPC) and Client Access.

How to Achieve Authority

QUESTION: How is authority over newly created objects assigned? How does the AS/400 know what the object authority should be for something that never existed before?

ANSWER: Authority other than that provided to the object owner and to *PUBLIC must be explicitly granted. The owner of an object has *ALL rights by default. It's where *PUBLIC authority is concerned that things get a little complicated, as there are three dif-

ferent places where the default *PUBLIC authority may be specified.

First, the Create Object (CRTOBJ) command itself contains an Authority (AUT) parameter, which permits *PUBLIC authority to be specified at the time of object creation. This parameter exists for all Create commands, regardless of the object's type. If the AUT parameter is not specified, the new object's authority comes from the Library create authority (*LIBCRTAUT) parameter associated with the object's library. You can view this value by using the Display Library

(DSPLIB) command.

The *LIBCRTAUT parameter may specify a particular value, such as *USE or *CHANGE, for *PUBLIC authority. Alternatively, it may "pass the buck" to the System Value Create Authority (QCRTAUT) value through the *LIBCRTAUT parameter value of *SYSVAL.

Ultimately, it is the QCRTAUT system value that determines *PUBLIC authority for an object. In specific cases, this value can be overridden either at the library level or for each individual object newly created through the CRTOBJ command.

Vincent LeVeque is a senior security engineer for **Science Applications International Corporation (SAIC)**. He can be reached at *vleveque@ earthlink.net.*

REFERENCES AND RELATED MATERIALS
- *Functions Reference* (SC30-3533-04, CD-ROM CO2E2001)
- IETF RFC Pages Web site: *www.ietf.org/rfc*