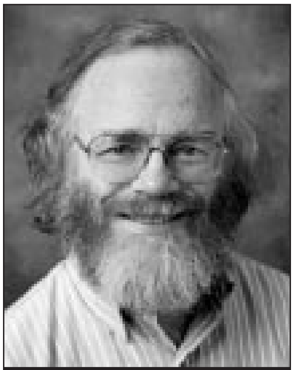
Wiretapping the Net

Steven M. Bellovin

To serve the needs of law enforcement while protecting privacy, the legal and technical approaches to Internet wiretapping must be reexamined.



Steven M. Bellovin is a fellow at AT&T Labs Research, where he does research in networks and security.[1]

Of late, there has been a great deal of interest in the ability of law enforcement agencies to place wiretaps on the Internet. While there are certainly legitimate reasons for wanting this ability, it is an area that is fraught with technical difficulty and legal ambiguity. In light of these problems, new legal and technical approaches to wiretapping are in order.

In this article, we examine wiretapping problems from several perspectives—statutory, jurisdictional, and technical—and suggest paths to minimizing those problems.

First, and in some sense the simplest, are problems with current statutes. While some of these can be fixed by legislative action, others raise deeper issues. Second are jurisdictional problems; Internet routing systems can make it unclear who has the right to tap a call and under what circumstances. Third, the very nature of Internet communications introduces complex technical problems. Packet-switched networks are inherently much harder to monitor than conventional phone lines. When looked at in combination, these factors highlight an overall complexity that makes the practice of Internet wiretapping a dubious undertaking.

Before discussing legal issues, it is important to understand, on a basic level, how the Internet works. The Internet is a packet-switching network; that is, a stream of data sent from one computer to another is split up into small pieces known as packets, and each packet is transmitted independently. Because each packet travels independently, all packets must be labeled with their source and destination addresses. These are known as IP (Internet Protocol) addresses and are not normally seen by end users.

New legal and technical approaches to wiretapping are in order.

Packets are transported over the Internet by a series of hops between devices known as routers, which serve as intermediate network nodes. A router looks at a packet's destination address and forwards it to the appropriate neighboring router. The most appropriate neighbor is determined dynamically by the routers—they regularly exchange information about the network topology by means of routing protocols. If two or more neighbors are equally good, a router will choose a neighbor so as to balance the load between alternate paths.

Each ISP (Internet service provider) operates its own network of routers. ISPs talk to each other via private links and at public exchange points where many ISPs meet. These interconnections, especially the public exchange points, are often overloaded, and the resulting congestion is a major source of data transmission errors, known as dropped packets.

There are no guarantees of packet delivery on the Internet—packets may be dropped, duplicated, damaged, or reordered along the way. All users who require reliable packet delivery must make arrangements for error detection and correction. Of course, most do, usually by using TCP (Transmission Control Protocol) in conjunction with IP. TCP detects damaged or dropped packets and sends acknowledgements back to the sending system when good packets are received. If the sending system does not receive the proper acknowledgements, it will retransmit the missing pieces until they arrive intact.

Applications that do their own retransmissions or that do not need all of the functionality of TCP sometimes use UDP (User Datagram Protocol). UDP is considerably lower in overhead and is often used for simple query/response applications.

Internet Addressing

In general, ISPs assign IP addresses to their clients on a dynamic basis. This is partly because addresses are in short supply and partly because IP addresses need to be flexible to reflect the current topology of the network. Consumers who use dial-up modems can end up connecting to the Internet through different routers (sometimes in other cities), depending on the load on their ISP's local modem pool.

Internet servers tend to have reasonably constant IP addresses, and they always have well-known names. These names are mapped into IP addresses via the DNS (Domain Name System). Apart from Web servers, ISPs run a number of servers on behalf of their users, notably e-mail and netnews machines. These servers are often replicated to provide load-sharing and reliability, and the duplicates are often geographically remote from the primaries. Similarly, corporate servers are often connected to more than one ISP, each of which could assign its own IP addresses to the corporate servers.

Internet connections are between pairs of systems, but performing user-requested services may involve intermediate systems. Consider, for example, e-mail sent between two typical home users. The mail is initially sent from the first home computer to the ISP's outgoing mail gateway. From there the mail is sent to the receiving ISP's incoming mail gateway, and from there it may be forwarded to a mail repository server. Finally, the receiving user dials in to the ISP, connects to the mail repository, and downloads the mail. At least three, and probably four or more, separate TCP connections are involved, as well as several DNS lookups. The multiplicity of systems involved in carrying out even simple requests is at the root of some of the legal complexities.

Any legal wiretapping has to be done in accordance with statutory authority. While in some sense this is an easy problem—the appropriate legislative bodies can simply enact any necessary laws—the problems of definition are significant. In particular, concepts familiar to law enforcement in the traditional telephony world do not necessarily translate easily to the Internet.

Statutory Considerations

The basic framework of U.S. wiretap law was adopted in 1968 as Title III of the Omnibus Safe Streets and Crime Control Act. (As a consequence, law enforcement personnel often refer to court orders permitting wiretaps as "Title III orders.") The law was significantly amended in 1986 by the Electronic Communications Privacy Act (ECPA), the primary thrust of which was to add the anti-eavesdropping protections—and the wiretap permissions—from the voice world to the data world.

As the law stands now, wiretrap permissions are governed by four major sections of statute: 18 USC 3121 (pen registers and trap-and-trace devices); 18 USC 2510 (interception of communications); 18 USC 2701 (access to stored communications); and 50 USC 1801 (foreign intelligence surveillance).

The pen register statute poses the greatest statutory problem. Pen registers are devices that record what telephone numbers are dialed *from* monitored lines, while trap-and-trace devices record the phone numbers that dial *to* a monitored line. These concepts were once well defined, but they pose considerable problems when extended to the Internet.

For example, what are the Internet equivalents to dialing and dialed numbers? The most obvious answer is the IP addresses in each packet, which represent the actual endpoints of the communication. But these endpoints are likely to be either uninteresting or well beyond the scope of a reasonable warrant. With a typical e-mail server system, all that could be learned by monitoring an end user's line is that the user is sending or receiving e-mail; the identity of the user is not disclosed. Similarly, putting the monitor at the ISP would reveal that a customer of one ISP was corresponding with a customer of another ISP. Neither party's identity would be revealed by monitoring at this level.

A more useful answer, from the perspective of the information to be learned, would be to monitor the actual e-mail addresses used. This, too, is problematic, for a number of reasons. First, e-mail addresses are not authenticated. It's quite easy to supply false source addresses by simply changing the mailer's configuration. This is a strong argument against trap-and-trace monitoring based on the sender's address. Similarly, destination addresses can be fabricated, and in this case, delivery failure notification would be e-mailed to the sender, and may not be detected by the monitor.

A more compelling problem is that the necessary user information may not exist, or it may be in the wrong place. E-mail sent via a "bcc:" option does not contain the recipient's name in the mail header lines, and typical mail retrieval protocols do not distinguish between mail header lines and mail content. Indeed, that lack of differentiation raises the most serious technical issue.

Internet standards have distinguished between the "envelope" and the "contents" of e-mail since at least 1982. In fact, those precise words are used. The envelope contains the instructions to the mail system about the sender and the recipients. With a pen register warrant, is it legal to look beyond the envelope? And if it were legal to look, the content in header lines poses two further problems. First, the information in a header does not necessarily relate to the actual sender or recipients of the mail. It is quite easy to manipulate a header to list addresses for people who will never see the mail, leading to the possibility of innocent parties being dragged into an investigation. Second, header lines may reveal important information about parties not covered by the court order—the addresses of other correspondents of the sender—thus going well beyond the capabilities of a traditional telephone trap-and-trace device. That is, if someone sends e-mail to the target of an Internet trap-and-trace order, the sender's identity will be disclosed to the investigators, as intended by the court order. But other addresses listed in the headers will be disclosed as well, despite the lack of any statute intending this result.

What are the Internet equivalents to dialing and dialed numbers?

A final problem to solve is the wording of the statute itself. Currently, a pen register is defined as "a device which records or decodes electronic or other impulses which identify the numbers dialed or otherwise transmitted on the telephone line to which such device is attached.” Similarly, a trap-and-trace device is defined as “a device which captures the incoming electronic or other impulses which identify the originating number of an instrument or device from which a wire or electronic communication was transmitted.”

Many of the technical problems with wiretapping stem from the very nature of Internet technology.

Note that both definitions not only specify a telephone line, they specify a “device” or a “line.” These concepts do not correspond to an identity, or even an e-mail account. At best, the analogs are a computer and either the access line or the IP address assigned to the computer. As noted, however, the latter is subject to dynamic change and is not particularly useful when dealing with e-mail.

The legal authority for full-content wiretap warrants is more clear. The ECPA amended the statute to speak of “electronic communications,” rather than just voice calls. Furthermore, the distinctions between IP address and destination, or between message envelope and content, are irrelevant; the investigator is entitled to all traffic. The difficulty comes in identifying the content belonging to the targeted user.

Most physical media used to carry Internet traffic are shared, with the exception (in some cases) of the access line to a customer’s premises. That is, the same physical wire or fiber-optic cable carries traffic to or from many different parties. To isolate a particular party’s packets, it’s necessary to look at the IP addresses. Is this sort of examination legal?

This question leads to questions of jurisdiction—Who has the right to place a wiretap, and under what circumstances? Does the physical presence of a packet in some particular locale matter?

Packets on the Internet can take a complex path from source to destination. This is partly due to the nature of IP routing, but even more to the complex relationships among ISPs. It is rather rare for a conversation to stay within a single ISP; just how and where they interconnect is governed by complex business and technical considerations.

A few recent experiments by the author make clear just how nonintuitive routing can be. In one case, packets between two towns in North Carolina went via Atlanta, Georgia. Packets from the author’s office to his home, both in New Jersey, went via California. And packets from New Jersey to Russia went to New York, Washington, D.C., and back to New York before finally heading overseas.

Access to e-mail raises more troubling issues. As noted, most e-mail addressed to individuals will reside on an ISP’s data center, on a mail server, until explicitly retrieved by the recipient. It is not likely that the data center will be in the same jurisdiction as either the sender or the recipient. What judge has the power to order access to such messages?

This problem is further complicated for international traffic. For example, in one test, traffic from North Carolina to Costa Rica went via Montreal. Does that give Canadian authorities any right to read it? For historical reasons, the United States is in the middle of many Internet paths. Does this give the United States the right to read such traffic?

Unreliable Packet Switching

Many of the technical problems with wiretapping stem from the very nature of Internet technology. Someone who wishes to avoid monitoring can exploit the complexity of the technology.

For example, one problem inherent to the Internet is that of packet stream reassembly. The individual packets that make up a message must be recombined at destination to form a coherent whole, and the rules for doing this are complicated. If the process is implemented differently on the monitoring box and the recipient’s system, the two might see different streams, especially if the target user attempts to evade the monitor. For example, consider two packets whose contents overlap in the final stream. This is acceptable to TCP, and in some cases is a normal occurrence; TCP compares the overlapping areas and ignores the duplicate content. But what if the two areas differ? Which packet should the monitor believe?

An attacker can make the problem even worse by exploiting packet lifetimes. Packets have a finite lifetime, measured in router hops. The “hop count” is assigned by the sending system, and each router on the path subtracts one hop. If the count reaches zero, the packet is discarded. Suppose there is a sequence of packets containing a login name, a set of backspace characters, and a different login name. Which login name is intended for the recipient? If the packets all have the same lifetime, the second one would be used. But if the backspace characters and second login name have too short a lifetime, the first name would be used. Can the monitoring system handle it properly? Using purely passive techniques, it is very difficult to tell how far away a destination is, and simply seeing different lifetimes on different packets says nothing about whether or not a destination will receive them.

Packets to a given destination can take different paths through the Internet. This can reflect topology changes or load balancing. Indeed, even a single computer can use multiple dial-up sessions in parallel to achieve greater throughput. A simple monitoring station may not be equipped to detect this. Furthermore, a very high percentage of paths are asymmetric: return traffic does not flow through the same routers as forward traffic.

Even deciding which packets to monitor is difficult. As noted, consumer machines generally have dynamic IP addresses. A monitoring station needs to know what IP addresses to watch, and that means it has to monitor the address assignment protocol, which can be difficult to do. If the monitoring station misses assignment messages, it will not begin to monitor the target; if it misses disconnect messages, it will record someone else’s traffic as well.

In fact, it is sometimes impossible to know what address is being used by a target. For example, some systems use Network Address Translators (NATs) to dynamically map a group of internal, private addresses to a few external IP addresses. Because of the shortage of IP addresses, some ISPs and many hotels employ NATs. A box monitoring a system on the public Internet has no way of learning the actual IP address of a correspondent system behind an NAT.

Software Complexity

The preceding description makes it clear that any monitoring system will of necessity be quite complex. That complexity carries with it its own risks—the most important being that complex software is buggy. It is generally accepted that the number of bugs in a system increases roughly as the square of the size of the code, and while bugs are never good, their consequences can be especially serious in an Internet wiretapping device.

The most obvious risk, of course, is that the device will crash. In some sense this is not so bad, in that such a failure is relatively obvious and benign, although it still represents wasted resources. More importantly, reliance on an Internet wiretapping device can divert investigators from the use of other techniques, and if the wiretap fails, no information will be gathered, by any means.

To make Internet wiretaps more accurate and secure, it is critical to limit vulnerabilities due to software bugs.

More subtle failures can have more serious consequences. Failures to record certain classes of traffic can easily deceive investigators; both exculpatory and incriminating evidence can be missed. Corruption in recording is worse yet. Apart from anything else, a recording that is demonstrably inaccurate is useless at trial, especially if it contains extraneous traffic.

But by far the most serious failure mode would be a takeover of the monitoring box by hostile parties, a scenario that is not at all improbable. About half of all new security failures are caused by “buffer overflows.” If a buffer overflow were to be found and exploited in an Internet wiretapping device, the device itself could be taken over, possibly by the target of the wiretap. And the consequences of that—the potential for criminal control of law enforcement tools—are chilling.

If nothing else, the attacker would be able to learn the monitoring parameters, and, depending on the design of the monitoring box, might be able to alter or erase logs of previously recorded sessions. Worst of all, the device—a system that by design is a high-quality wiretapping unit—could be diverted and used as an eavesdropping unit for the attacker.

We have outlined a number of difficulties involved with wiretapping on the Internet. While some of the problems are very hard to solve, we can address some issues in a number of respects.

The first approach, of course, is to clarify the statutes. While the ECPA was a good first try, experience has shown that it does not match the reality of the Internet. The problem of pen registers, in particular, is a thorny one, given the inherent difficulty in determining the endpoints of the conversation—the target and the target's correspondent—in a way that cleanly separates that information from the content of their communications.

A second approach is to push the wiretap as close as possible to the target. Much of the trouble arises from differences between what the user sees and what the monitoring box sees. Other problems come from identifying just which packets belong to the user. A modem tap on the physical phone line would finesse many of these issues; packets on that wire are, by definition, to or from the user. Questions of which IP address to monitor are moot. Solutions of this nature (placing the tap near the target) could also be applied to DSL (digital subscriber line) connections, but not to cable modems, which are inherently shared.

Finally, simplifying the task to be performed will simplify the software. (That is, if the tap is closer to the target, there are fewer variables to handle with software. Among other things, it's much easier to monitor a slow line than a fast, multiuser cable.) Software complexity is the greatest unsolved problem in the computer industry and is likely to remain so. To make Internet wiretaps more accurate and secure, it is critical to limit vulnerabilities due to software bugs. Only in this way can we be confident that both the restrictions and the authorizations of the law are carried out.

Notes

[1] The author recently served on a committee of the National Research Council that produced the report *Trust in Cyberspace* (National Academy Press, 1999). The Committee on Information Systems Trustworthiness operated under the NRC's Computer Science and Telecommunications Board.