Wireless Networking Considered Flaky

Eric Allman, Sendmail

You know what bugs me about wireless networking? Everyone thinks it's so cool and never talks about the bad side of things. Oh sure, I can get on the 'net from anywhere at Usenix or the IETF (Internet Engineering Task Force), but those are `_hostile_nets_`. Hell, all wireless nets are hostile. By their very nature, you don't know who's sharing the ether with you. But people go on doing their stuff, confident that they are OK because they're behind the firewall.

Let's face it: WEP (Wired Equivalent Privacy) is a joke. There's no privacy on a wireless net. When you type your password, it's there for the world to see—and take, and abuse. A lot of places don't even bother with WEP, even behind firewalls. You want free 'net access? Drive into a random parking lot in Silicon Valley and pull up next to one of those big, two-story "ranch house" style buildings that seem to be ubiquitous there. You'll have a shockingly good chance of being on the 'net. But not just the Internet: their `_internal_` network. And if you sniff that network you might just get a password or two. Or maybe several dozen. You'll probably even trip over some root passwords.

And what about bandwidth? Eleven megabits? I think not. That's a theoretical maximum, not anything you're likely to ever see. Does anyone today still think you can get 100 Mbits out of a clearly labeled 100-Mbit Ethernet? I'm running one at home, and I sometimes get as much as 20 Mbits out of it. But usually not. Normally I get somewhat less than 10 Mbits from it. That's an order of magnitude of wasted bandwidth. The only time I've ever heard of an ethernet getting anything close to full bandwidth was in a lab. On a private network. With two nodes on the net. With specialized IP stacks. One node transmitting and one node receiving. Zero contention. And massive MTUs (maximum transmission units), far beyond the ethernet spec. And even that only got about 80 percent of the bandwidth.

OK, so let's assume that in the real world you get about 20 percent of the theoretic bandwidth from a contention network. That's probably acceptable, because that's one host to one host; there may be some idle chatter going on, but basically my home network is pretty quiet most

There's no such thing

AS SECURITY OR STABILITY

ON A WIRELESS NETWORK.

BUT WE USE IT EVERY DAY.

of the time. If I had more traffic going on I would probably be able to use more of the bandwidth, because a lot of that delay

is in application and TCP stack overhead. So with another pair of hosts talking at the same time, maybe I would be able to use, say, 35 percent of the bandwidth. And that's pretty good, considering that any ethernet running at as much as 50 percent capacity is probably on the verge of utter collapse.

But with wireless you have all sorts of other problems. Did you know that the reason the 900-MHz band is free is that it has some wee technical problems? Like that's the harmonic frequency of water. As in, moisture in the air. Have you ever noticed that your wireless network doesn't perform as well when it's raining? And buildings. Especially buildings like hotels, where Usenices and IETFs are held—hotels that have lots of steel beams holding up 30 stories of building. At one IETF I was sitting in the lobby, within eyesight of the base station. Eyesight, right, as in "line of sight"? To a base station that was maybe 10 meters away from me. And I couldn't get a signal. But if I moved one seat to the right I had four bars. Which I would have done if it weren't rude to sit in someone else's lap. At least before being introduced. At home I have semi-dead spots where moving my laptop 20 centimeters one way or the other makes the difference between no signal and a usable signal. And it's not like it's stable: The signal can disappear and reappear a minute or so later even without moving my laptop at all. And this is even on dry days; if it's raining all bets are off.

Can you fix this stuff? Some of it, sure. I would think I wouldn't have to tell anyone at IETF that the world is a hostile place and maybe they should use Secure Shell (SSH) instead of telnet, and never, ever type their passwords in the clear. But at every single IETF, at every Usenix, at pretty much any conference, passwords are sniffed, literally out of thin air (sorry, I couldn't resist). Lists of passwords get published on the message board, not to be hostile, but just to let people know that maybe,

Continued on page 95

just maybe, they should think about changing their passwords. And maybe think about encryption. Of course, that's the good guys who publish the passwords on the message boards. The bad guys, who you have to assume also have the passwords, are using them themselves or posting them on bathroom walls somewhere.

I run SSH. When reading mail, I use an encrypted IMAP (Internet Message Access Protocol) port. When sending mail, I use TLS (Transport Layer Security). I try to make sure that the only thing that goes through my network interfaces unencrypted is basic Web browsing, where by "basic" I mean "no passwords or other confidential data are exchanged."

But some problems can't be fixed. The laws of physics just aren't going to be repealed anytime soon. Wireless networks are going to be flaky; that's just inevitable. And as much as I hate it, I use it. Every day.

Just don't tell me that it's a perfect world.☹

LOVE IT, HATE IT? LET US KNOW:

feedback@acmqueue.com or www.acmqueue.com/forums

ERIC ALLMAN is the cofounder and chief technology officer of Sendmail, one of the first open source-based companies. Allman was previously the lead programmer on the Mammoth Project at the University of California at Berkeley. This was his second incarnation at Berkeley, as he was the chief programmer on the INGRES database management project. In addition to his assigned tasks, he got involved with the early Unix effort at Berkeley. His first experiences with Unix were with 4th Edition. Over the years, he wrote a number of utilities that appeared with various releases of BSD, including the `-me` macros, `tset`, `trek`, `syslog`, `vacation`, and, of course, `sendmail`. Allman spent the years between the two Berkeley incarnations at Britton Lee (later Sharebase) doing database user and application interfaces, and at the International Computer Science Institute, contributing to the Ring Array Processor project for neural-net-based speech recognition. He also coauthored the "C Advisor" column for *Unix Review* for several years. He was a member of the board of directors of Usenix Association.

© 2003 ACM 1542-7730/03/1000 \$5.00.