

Scientists and Hackers

David Warsh

July 25, 1999

For example, as one of a series of recent ads, IBM Corp. has been parading a photograph of employee Nick Simicich, "Ethical Hacker". Flowing-bearded, fish-shirted, felt-hatted, granny-glassed, Simicich is described as a "paid professional paranoid". His job : to determine whether software systems are susceptible to hackers. His distinction : he protected the electrical power infrastructure in the United States from "cyber-jackers bent on misdirecting electricity with bogus information".

So much for the limits of the current definition of hacker : the principled tester-of-limits vs. the high-tech saboteur. But if you listen carefully to the leading innovators in computer systems, you realize that the applicability of the term is expanding rapidly. Consider the story of how Linus Torvalds, and not Richard Stallman of the Massachusetts Institute of Technology, came to pose the chief threat to the world's first centibillionaire, Bill Gates.

Stallman would seem to be the hacker *par excellence* - "the last true hacker", as author Steven Levy called him in his famous book. Even though it was Harvard College that he entered in 1970, it was down the street at MIT that Stallman found his spiritual home. There he imbibed deeply the peculiar sensibility that is one of MIT's enduring contributions to computer culture. He rendered a small jewel of a hack of one sort a few years ago when he posted an official-looking sign on the spot that serves as the Wellesley terminus of the shuttle bus that has connected Wellesley College and MIT since the mid-1960s. Barbedall Square, it read. At the MIT end of the line, of course, is Kendall Square (it helps to say them out loud). Such is the world of an MIT hacker : high-spirited, clever, accomplished, and, just possibly, but not necessarily, oriented to the occasional countercultural prank. According to Stallman, the use of "hacker" to mean "security breaker" is mass media confusion. Hackers themselves, he says, use the word to mean "someone who loves to program and enjoys being clever at it".

Stallman's significance goes far beyond a few practical jokes, however. It was in 1981 that a little start-up company known as Symbolics Inc. hired away most of the staff of MIT's Artificial Intelligence Lab and set them to work writing proprietary software. At about the same time, Digital Equipment introduced its new VAX computers, with brand-new proprietary operating systems. The community of pioneering software sharers that had grown out of the Model Railroad Club at MIT suddenly collapsed. So Stallman did a historic thing. Instead of joining the proprietary world, he set out to make a free, shareable operating system that would work on *any* computer and run any program. There would be none of the nondisclosure agreements that characterized the nascent software industry.

Stallman dubbed his ambitious project the GNU system. But first there would have to be tools. His first big achievement was EMACS, a compiler and text editor that rendered possible more ambitious programming. Other programs followed. All were freely made available to others under the "copyleft" license Stallman and his friends devised to keep the underlying source code open - that is, to protect the right of other users to know and modify the basic code. Stallman's Waterloo was the operating system's "kernel", the core part of the operating system where memory is allocated among all the other programs : scheduling, signalling, device input/output, and so on. When Stallman turned his shoulder to the task in 1990, the dogma among computer scientists in cutting-edge American universities such as MIT was that something called a "microkernel" would be required to make a truly portable system. A microkernel would be a kind of coded general theory of all computer architectures; it would be required before such a system could run on any hardware.

Cut to Helsinki in 1991. A young Finnish graduate student named Linus Torvalds, comfortably outside the circuits of grant-supported American computer science research, decided to try an alternative approach - a "monolithic kernel", simpler, but far faster and already relatively well-understood. "I am a pragmatic person", Torvalds has written. "... I didn't have to aim for such a lofty goal. I was interested in portability between real world systems, not theoretical systems". So Torvalds read up on the systems in use, in search of common denominators between them. Once he had a design for a certain task - for memory management, say - that would be on the most popular chips, he put it out to an extensive list of correspondent hackers to see how it could be improved. At first it was written to suit just one architecture : the Intel 386. Gradually a kernel emerged that could control the most popular microprocessors - the 68K, the Sparc, the Alpha and the Power PC. Torvalds then combined his kernel with a good bit of the GNU programs Stallman and his friends had written, and presto ! The operating system that has become known as Linux - similar in spirit to AT&T's Unix system but not based on it - was ready to be distributed and more or less continually improved.

And because it had been written - hacked - by an impassioned graduate student in Finland and a relentless code warrior in Cambridge and a few hundred collaborators for their individual satisfaction and shared use, it was available to others for free. And in the last 10 years the project originally envisaged by Stallman and Torvalds and a handful of others has grown into a credible threat to Windows NT - the Microsoft operating system with its secret proprietary source code on which rests Bill Gates' most basic hopes for the 21st century.

"Linux today has millions of users, thousand of developers, and a growing market", Torvalds has written in *Open Sources : Voices from the Open Source Revolution*, the O'Reilly & Associates anthology from which this account is drawn. "I'd like to say I knew this would happen, that it's all part of the plan for world domination. But honestly this has all taken me a bit by surprise. I was much more aware of the transition from one Linux user to one hundred Linux users than the transition from one hundred to one million".

And the point ? Simply that the neatly barbered and quietly circumspect Torvalds is every bit as much a hacker as Stallman - a fact the flamboyant Stallman readily concedes.

So what is a hacker, after all ? Eric Raymond offers this definition in the third edition of his *New Hacker's Dictionary* : "A person who enjoys exploring the details of programmable systems and how to stretch their capabilities, as opposed to most users, who prefer to learn only the minimum necessary". Torvalds' success suggests that even that definition may be too narrow. Remember, it was only 150 years ago that thinkers and wordsmiths of all sorts were trying to agree on a term that could apply equally to all the different sorts of intellectuals who had emerged from the precincts of philosophy and natural history. It seemed clear these new professionals shared an ethic. Their methods and goals were unfamiliar and, quite possibly unique. They even admitted women to their ranks ! In the England of the 1830s, it took a decade before the term "scientist" emerged and won common acceptance.