Interview with Dan Farmer

RIK FARROW

Dan has been the security architect for four Fortune 500 companies, started his own enterprise software company, and has researched, written, or coauthored a variety of security software tools, papers, essays, and a book (often with his erstwhile colleague Wietse Venema). Someday he'll get that research gig, even if it's after retirement. zen@trouble.org

Rik is the editor of *;login:*. rik@usenix.org

I first met Dan Farmer during DEFCON 1, where I thought he had the most useful and interesting presentation there. I had heard of Dan because he had written COPS, a very early, if not the earliest, vulnerability scanner. I kept encountering Dan over the years at various USENIX conferences as he continued to write tools, papers, and work on improving Internet and *nix security. I also met Wietse Venema for the first time when Wietse and Dan were presenting their forensic toolkit in 1999 [1].

Dan has often appeared in the limelight, partly because he feels so strongly about the general lack of security, but also out of a deep sense of ethics (and contrariness) that has guided his life, often at the expense of his career.

Rik: How did you first get interested in security?

Dan: Growing up I used to love the spy vs. spy mentality in movies and books, but outside of government work there didn't seem to be any way to make a living at it, and I'm afraid I wasn't cut out to live inside the Beltway. But right before I graduated from Purdue there was a massive international security event called the Morris Worm [2]. The network was getting slammed and people were running around in the halls trying to figure out what was going on.

For whatever reason, the Worm captured my imagination like nothing had done before; if computers could do this, there was hope yet. The following semester was due to be my last, but aided by the fact I was a not a good student, I had to take one last course over the summer. Gene Spafford was a professor there and had written one of the two important papers on the Worm, so I simply walked into his office and said I was interested in security and was there any sort of coursework I could do over the summer that would help me graduate.

He agreed to do a special course; I told him I'd like to write something that would test the security of computers, and we worked out the basics of a security tool. It felt like the first time in my life that I had a purpose; I was simply beyond joy to be working on the project. Looking around today it might be hard to imagine that there was just about zero written about security; after months of searching, I had found one book (*UNIX System Security*, by Wood and Kochan), a very small pile of articles, and one jewel, Bob Baldwin's MIT master's thesis that detailed an expert system that probed system security (Kuang). I cobbled together everything I could into one program, which I named COPS, and put it out on the Internet for free.

I thought that that was pretty much it for my security career, but in hindsight my timing was nearly perfect—since there was so little information and zero programs out there, people started assuming I was some sort of security expert rather than an obsessed young lad; and after giving a USENIX paper on COPS [3], I was offered a job at CERT, which was created after the Worm.

It wasn't until many years later that I was able to thank Robert Morris for writing it and starting my career.

Rik: What was it like working for CERT in its early days?

Dan: I think I was the sixth or seventh person hired on at CERT—they had four sharp technical folks already there, and Rich Pethia (a great guy) was leading the charge. Amazingly, a quarter century later, he still manages the group there.

Unless you were around at the time it might be hard to even imagine the paucity of security knowledge then. CERT was a radical idea—set up a 24-hour hotline for anyone in the world to call if they had a security problem, question, or concern. We were one of the only places in the world outside of some tech-savvy governments that knew much of anything about Internet security.

When people started sending us information about break-ins it was a revelation—international intrigue, companies, universities, governments, militaries all over the world getting broken into.

CERT was a good place to work, but I wanted to start researching the latest and greatest, which at the time were network worms and malware. CERT was created by a worm, so why wouldn't they want someone looking at them more in depth? After all, as Sun Tsu reportedly said, to know your enemy you must become them.

Needless to say, they didn't quite agree, but one of my personality defects is my almost pathological contrariness. If people tell me to stay away from something it's something akin to dropping a cardboard box in front of a housecat, we'll both hop right in. So when Sun went looking for a technical head thug in their newly founded security team, I headed for the West Coast; nearly twice the salary didn't hurt either.

Rik: What was it like working in Silicon Valley?

Dan: Silicon Valley was the place to be in the '90s; the Net was exploding, companies zooming up the Fortune 500, and then along came this company called Netscape, and with real money on the line, security started getting the tiniest bit of respect, or at least some modest afterthoughts.

I met Brad Powell in my first stint at Sun in the early '90s; he's a huge guy with a big heart who saw his Biathlon Olympic dreams dashed when Jimmy Carter boycotted the 1980 Olympics because of the Soviet invasion of Afghanistan. Brad wrote a TITAN prototype because he'd run COPS on his customers and was getting really tired of fixing by hand all the myriad security problems COPS would routinely find. We kept in touch after I left, and after haranguing him for years to release his code, he finally agreed if I'd help him spruce TITAN up. TITAN didn't just scan for vulnerabilities, it would also repair them. Brad was a great security guy but at times bore the curse of the engineer and wasn't able to articulate things to mortals.

So I re-architected TITAN, made it possible for normal people (well, normal system administrators!) to actually use the thing, and reassembled and amplified his words to create the USENIX paper [4].

Rik: Where does SATAN fit into the time frame?

Dan: After seeing real incidents at my time at CERT, I became really interested in how people were breaking into computers. At the time security—and especially things like bugs and break-ins and such—were not discussed in polite company, and I couldn't find anyone who had any information at all about how people or programs actually compromised systems. So I sat down and wrote up all the different ways I thought it could happen.

Fortunately, while working at Sun, that fit perfectly into my job, and I had a playground of many thousands of systems that I could legitimately break into. I remember one winter vacation breaking into then-CEO Scott McNealy's workstation and all but one of Sun's 50 VPs' workstations (after asking permission, of course), and only missed that one because the VP had apparently turned off his computer over the break. Fortunately, McNealy was really good-natured when I told him about it in the hallway in passing.

But I didn't feel I had the technical firepower to put out a paper on various ways to break into (and protect) computers on my own, so I reached out to someone I'd never met, a Dutchman by the name of Wietse Venema. Wietse had created the best security tool that had ever been written, TCP Wrapper [5], and seemed perfect for the task—if he wasn't, perhaps he would know who would be. Fortunately, Wietse was intrigued by the project, and we coauthored my favorite project, which detailed how to break into computers along with various defenses you could use to protect yourself.

For some reason we really hit it off and remain close friends to this day. Wietse remains an intellectual with astonishing programming skills, and I was the crazy dreamer who would try to convince him that something utterly ridiculous was a good idea. Postfix, his wonderful mail project and what most people identify him with, was all his idea. Although Wietse dismissed my warnings that it'd take a lot longer than he thought it would and he'd be chaining himself to it for the rest of his life if he went through with it, it all turned out well.

In any case, we mentioned in the paper [6] that we were working on a program called SATAN [7] (Security Analysis Tool for Auditing Networks), never dreaming that people would actually care. However, the paper, the name, and the promise of an automated security scanner struck a chord with our audience, so we started thinking more deeply about how to actually do it.

Perhaps to the disappointment of our audience, we spent the next couple of years writing, talking, and traveling to visit each other; SATAN was perhaps the first security vaporware, the Duke Nukem Forever of its time, until we released it on my birthday in 1995. The delay fueled excitement and anticipation, and the pundits and press had a field day about it all. Fortunately, the Internet survived and it wasn't quite like "randomly mailing automatic rifles" to people or other colorful quotes that found their way into the media. Perhaps the best part of the program was the browser-based UI, which I think was the first of its kind.

Rik: Didn't that result in you losing a job?

Dan: A few months before SATAN's birthday I had gotten a job as the Security Czar of Silicon Graphics (SGI). I still didn't anticipate the fervor to come, but I made sure my boss knew before I was hired of the program and its possibly provocative name. Just prior to the planned release I was called into a meeting, and found myself alone with a vice president and a couple of lawyers, who claimed no prior knowledge of my work or plans for SATAN. I didn't immediately catch on, but soon did after they gave me some options; I could release the program to SGI's customers, I could simply not release it at all, or I could work with SGI to make it a product. Or I could walk.

Another character flaw of mine is saying what I think rather than perhaps being a bit more politic, so I refused their offers to take off with our work and never set foot at SGI again.

Rik: What happened next?

Dan: I went back to work for Sun and was able to do some refueling and research. Along the way I tossed an idea offhandedly to my boss about centralized security management and monitoring; he was pretty stoked about it and asked if Sun could use it for commercial purposes. I replied that I'd have no problem at all with that, but I wasn't going to work on it if it were productized. I had no interest at the time in working in engineering or for a Sun product line.

After helping get a prototype built, we showed it to Eric Schmidt (who later moved on to far greater fame and glory at Google), who gave the order to productize it, and that I'd be the one running the product show. Shortly thereafter I had a conversation with my boss, he said he remembered our deal, but he was ordered to put me in charge…so I quit.

Rik: Jumping ahead to a few years ago, I understand you got some DARPA research money for a security project. Tell us about that.

Dan: My old pal Mudge (Peiter Zatko) had been running around DARPA for a bit and had helped created something called the Cyber Fast Track program. Mudge had been haranguing me into submitting a proposal for it. Coincidentally, I'd just been laid off from being Symantec's security architect when they dissolved the entire architecture group as part of a further move towards outsourcing; dozens of us were put out on the street pretty much the same day. Armed with some free time, I submitted a proposal pretty much as a lark; Mudge's claims were so outrageous that it seemed doubtful anything would come of it.

To my surprise, it worked exactly as he claimed. You could submit a small writeup (some two dozen pages at most) about pretty much anything you wanted to work on in security for some months, and within seven working days the US government would say yes or no. I didn't think our government could decide on the time of day in that short a time, let alone grant a contract to work.

Perhaps my favorite work in the Fast Track program was researching IPMI [8], a rather obscure and, as it turns out, insecure out-of-band management protocol that servers speak. The actual project was just a few months of work, but I got intrigued and spent nearly all my free time on it. I ended up working with Fast Track for a couple of years and would be happy to continue similar research, but the program is over now; all things must pass.

Rik: Anything else you'd like to say?

Dan: Over the years, as the tech field has gotten more mature, it seems as though they've squeezed out a lot of the freedom and innovation that fueled the Internet, and more than ever it's simply the financial numbers that matter. Obviously, the numbers do matter, but I don't think it has to be at the expense of everything else. I've no regrets, but if I'd known how it was going to turn out I probably would have gotten that PhD along the way, as the lack of the PhD pretty much excludes me from any institutional research areas, which is where I probably should have gone.

I've been asked many times what they should do by people wanting to get into, or get ahead, in the security business. For me the answer is always the same—follow your heart and give back to the community that helped you get where you are today. This is one of the reasons open source is so important.

For me the hardest thing to do is to keep putting your work and self out there—after all, what the heck do I know compared to all these incredibly smart and capable folks (especially the young ones) who already know computers better than I ever will?

I hope all of this doesn't sound like I'm ungrateful, because I've been extraordinarily fortunate that I've been given the opportunities I've had. I've been called a security expert pretty much the day I got my job at CERT, but I'm pretty dubious about the title—mostly I just had good luck getting into security before most.

References

[1] The Coroner's Toolkit (TCT): http://www.porcupine.org/forensics/tct.html.

[2] The Morris Worm, a historical view: http://www.washingtonpost.com/blogs/the-switch/wp/2013/11/01/how-a-grad-student-trying-to-build-the-first-botnet-brought-the-internet-to-its-knees/.

[3] D. Farmer, E. Spafford, "The COPS Security System Checker," USENIX Summer Conference, June 11–15, 1990.

[4] D. Farmer, B. Powell, and M. Archibald, "TITAN," LISA '98: https://www.usenix.org/legacy/publications/library/proceedings/lisa98/full_papers/farmer/farmer_html/farmer.html.

[5] TCP Wrapper: http://en.wikipedia.org/wiki/TCP_Wrapper.

[6] D. Farmer and W. Venema, "Improving the Security of Your Site by Breaking into It": ftp.porcupine.org/pub/security/admin-guide-to-cracking.101.Z.

[7] SATAN: http://en.wikipedia.org/wiki/Security_Administrator_Tool_for_Analyzing_Networks.

[8] Dan Farmer's IPMI research: fish2.com/ipmi/.