

THE LONELINESS OF THE XZ TOOLS DEVELOPER

THE HISTORY

The other day the news of an attack on Linux appeared. The developer of XZ Utils, (compression algorithms used in many parts of Linux), did it in his free time and without receiving compensation.

The man felt pressured, more and more requests, and with urgency. The man reported his problems (work overload, feeling burned out, and psychological problems). An assistant appeared, who worked with him for 2 years. But the pressure was growing, and no one did anything for him. So, at the assistant's insistence, he permitted him to access the code repository.

The assistant prepared a new version, in which he introduced a backdoor that allowed escalation to administrator privileges.

The malicious code was discovered by chance by a Microsoft engineer while testing SSH performance. The assistant disappeared and they don't know who he is or who he worked for

Everyone throws up their hands thinking about what could have happened. But in an exercise of shameless hypocrisy, no one has looked at the person who has been supporting him for years without recognition or remuneration.

In this article, I want to talk about the developer and the Free SW system.

I am a vulgar man, that makes me think and feel like most people when faced with the same situation. I don't know the developer of XZ Utils, but I am going to talk about my personal experience as a Free SW developer, through it you will see him and you will see the problem from his perspective, which it seems that no one wants to take.

I am a developer of the Boost Sort library, in C++. ([link to Boost Sort](#)). The library is composed of 7 algorithms, 4 single-threaded and 3 parallel algorithms. I am responsible for the 3 parallel algorithms and two single-threaded ones, and I am also the inventor of two of them. I am currently working on the internal reconfiguration of the parallel algorithms so that they automatically adapt to the HW on which they run (a micro-controller with 4 cores and little memory is not the same as a server with 2TB of RAM and several hundred cores) and the data type to sort. I am also implementing two new parallel algorithms that I have invented, to improve the speed and reduce the memory consumption of the existing ones.

THE WORK OF A DEVELOPER

The first thing you need to become a project developer is competence. Be able to understand, develop, modify, and correct code. It is hard, complicated, and often laborious work, requiring many hours. (Taking someone else's code, understanding it, and then modifying it is something that has made many of us sweat.)

Why do I do it? A certain sense of duty, I help others, because others have helped me before. And since I have a programming talent, I use that talent to help others. In addition, some people asked me for help and I promised to help them. I gave them my word, and for me, although it sounds old, it is sacred.

If it's your job, and you get paid for it, you have 8 hours a day. If not, you do it in your free time, and at most, you can get one or two hours a day at most. (In fact, I have missed many things that I wanted to do, because that code had to be fixed, and it was important and urgent). This means that in 1 week you do what you would do in 1 day dedicated to it. And a week dedicated to it full-time is more than a month of my work.

Another factor to take into account is fatigue. Think that that hour or two hours is in addition to the 8 hours of work for which you are paid a salary. We must not lose sight of the fact that in addition to work, you have a partner, and children and that developing Free SW does not exempt you from your part in household chores. Many times, the time dedicated was the night when everyone had gone to bed, and you were tired but also calm.

You are alone, and what you receive are requests to do something, often urgently, and occasionally an acknowledgment or a simple thank you from someone who uses it. These small recognitions are important, because there are times when you are in a very low mood, or you are busy with a multitude of small issues, which do not allow you to take time to maintain the code.

In those moments of low mood, the idea of sending everything to hell arises strongly and frequently. And those small recognitions make you feel that your effort is useful and that you are helping people you don't know, but who trust you and your work.

THE ADA EFFECT

Most of us free SW developers are no longer 50. Linus Torvalds recently warned that in 10 or 15 years, many of the developers will not be around and that is a problem.

I am from a generation that learned programs on their own. At University I learned some BASIC and Z80 assembler. All the other languages that I have learned since (Pascal, COBOL, Fortran, C, x86 Assembly, C++, Ada, Java, Rust, Python...) no one has taught me.

We were a generation voracious to learn, and irreverent towards the prestige of IBM and Digital VAX. They derogatorily called us brats, but the future was ours.

With the first PCs (8088 at 4.77MHz with 96 K of RAM, without a hard drive and with two 180 K floppy drives) we did wonders. My final degree project was a program for solving systems of differential equations for the calculation of electronic circuits, in Basic, with one of those first PCs that arrived in Spain.

I got a Turbo Pascal and learned Pascal, and I got a Turbo C and learned C. But when I wanted to learn ADA, it was very difficult to get a compiler for PC. ADA was a very “prestigious” language and could not be left available to everyone.

Young people grow up and when they go to companies and have to develop SW, they do it in the languages they know. I programmed the first TCP/IP that arrived in Spain in 1988, for a test bench for the McDonnell Douglas F-18 Hornet fighter aircraft, between a microVAX and a Harris 800. One part is in assembler and another in C. I also programmed the first Internet consultation that was made in the Ministry of Education of Spain with my Turbo C++, I programmed the CGI protocol and the applications that invoked it.

When those responsible for ADA realized that there were no ADA programmers, they created the GNAT project, which is an excellent ADA compiler. But his time had passed. We were in companies and we had families, and the next generation did not have our voracity. This is the most important reason why an excellent language (ADA) has been limited to limited areas and few people use it to program (I have been a member of ADA Spain since many years ago).

THE FAMOUS PROPERTY OF TOILET PAPER

The same thing happens with the Free SW. Those of us who are here are, for the most part, from that previous generation, but our end is already visible. But if something is not done soon, it will happen that many projects will be left without their developers and spare parts, so what happened to ADA could happen to Free SW.

Will they be able in the future to recruit people with high technical knowledge, with altruism to do complex work without pay, and with enough resilience to work alone and under pressure? Thinking that they are going to appear because they have appeared before can lead to disaster for many projects on which current computing is based.

And we must not forget that the developers of the Free SW have the famous property of toilet paper. When you miss them, it's already too late.

Francisco Tapia
fjtapia@gmail.com