

Re: What's the story with the "end of XP"?

Source: <http://coding.derkeiler.com/Archive/General/comp.arch.embedded/2007-06/msg00691.html>

- *From:* David Brown <david@xx>
 - *Date:* Mon, 18 Jun 2007 16:55:23 +0200
-

Chris Hills wrote:

In article <467664c6\$0\$1440\$8404b019@xxxxxxxxxxxxxxxx>, David Brown <david@xx> writes

With an open source product, you are much safer. *I* may not understand much of the source code for GPG (an open source alternative to PGP), but I can be confident that there are plenty of people around the world who *do* understand it.

I doubt that.

What exactly do you doubt? I presume it's the final statement – that there are plenty of people around the world who understand the PGP code. I made that statement on the knowledge that there are cryptography experts around the world, employed in the private sector, employed by governments, and in the academic world. You can't call yourself an expert in cryptography without a solid familiarity with some of the most used algorithms and implementations – programs like GPG, OpenSSH and OpenSSL are known to be among the most secure solutions precisely because there are so many interested and capable people who have studied their source code.

Note that I am not simply repeating the mantra that "with many eyes, all bugs are shallow" – a concept that is only true in specific circumstances.

If flaws are found, they will be fixed quickly – open source developers can't hide flaws in the same way closed source developers can.

This is also not true.

Again, I must ask what you think is not true. There is overwhelming statistical evidence that serious flaws in major open source software projects are fixed quickly – generally far faster than in comparable closed source projects. There is also little doubt that in a closed source project, it is far easier to hide flaws – if I find a bug in a program I have written, I can simply keep the information secret (perhaps revealing it when I have a fix

Re: What's the story with the "end of XP"?

ready, perhaps never revealing it). When a security researcher finds a flaw in an open source project, everyone can hear about it.

All your arguments previously seems to imply there is a difference between closed source and open source programmers.

There is a difference in the programming methods – not necessarily a difference in the programmers themselves. Many programmers do both.

When you are looking at embedded development tools, the situation is a little different – there is a much lower ratio of users to developers, and the users themselves are programmers. Here the source code can be very useful – I've often found source code useful to understand what is going on, or while debugging difficult problems.

I agree. But it does not need to be FOSS. Most of the commercial SW I know where having the source would be useful it is provided.

I don't think I've made any claims here about licensing – just about the availability of the source, along with the ability to modify and re-compile it yourself. As I say, there are differences in the requirements for embedded tools and for general purpose software. When I am doing embedded development, it is important that *I* can read and change the code running on the target. When I am looking for a trustworthy application for safely accessing the internet, I want to be sure that other experts can read and modify the code, and that I can benefit from that. In the first case, the source license can be more restrictive, while in the second case, a free (as in speech) license is essential.

I'm not too bothered about having source code for a compiler or debugger

But the rest of the FOSS brigade seem to thing it is essential.

That sounds like prejudice and overly wide generalisation to me.

There are certainly people who feel that the all software should be open source, and that anything else is evil. And there are people who feel that anyone can use whatever license they want, but that they personally will only use FOSS. There are others who will use FOSS where possible, and closed source when they have to. Personally, I try to pick the best software for the job at the time, taking into account many aspects of "best". Thus I am writing this using Thunderbird, running on Windows, connected to a Linux server. In my work today I have used two open source compilers and one closed source compiler. I have used an open source browser and an open source office suite, and a closed source windows IDE. I mix and match as appropriate.

Re: What's the story with the "end of XP"?

Re: What's the story with the "end of XP"?

For most programs, being open source is a benefit IMHO, although it is not often the most important reason to chose a particular program.

Looking at your case where the customer attempted to fix the software flaw themselves, there are a number of ways to view it. You could say that the customer made a mistake by trying to fix the compiler themselves – tools like compilers are pretty specialised, and it is hard for an outsider to jump in and make successful changes.

Not according to many FOSS devotees we get around here. Though strangely the commercial compiler writers I know would agree with you.

Any FOSS devotee will understand that it is hard to make successful changes to a compiler or tool of similar complexity (although open source projects are often easier to work with, since they are built as collaborations in the first place). What a FOSS devotee will tell you is that it is important (to them) that making such changes is *possible*.

But it's important to consider what might have happened, especially if they did *not* have access to the code. The supplier could well have claimed there was no bug, they could have decided that fixing the bug was low priority and would be done in the next version, they could have agreed to fix the bug but taken too long, they could have been bought up by a rival company and stopped all product development – there are any number of plausible scenarios where the customer would have been in big trouble. Having the source code themselves gave them an escape clause – they could always get out of the problem, albeit at a cost. It turns out that in this case, with hindsight, fixing (or trying to fix) the problem themselves cost a lot – but it was perhaps a sensible risk management decision to maximise the chances of having working software in time for delivery. Having the source code available is not a cost or a burden to the customer – it gives them more freedom and more choice.

I am not convinced. I have seen cases where it was a a hindrance.

That goes along with an argument against giving developers fast computers, in case they spend their time playing games instead of working. Giving people more options is only a hindrance if they can't deal properly with making decisions about priorities in their work.

There does seem to be an implied line in most of the FOSS rants. They usually claim that commercial pay ware is buggy and thereby imply that Sw Engineers working for these companies are less capable than the FOSS community. This always amuses me especially as many of them are actually paid to write SW for some one else.

Re: What's the story with the "end of XP"?

Again, that is blatant prejudice – you've been listening to too many fanatics. I know that some fanatics can be loud (it's the case in any argument), but you are allowed to think for yourself, and to listen to other people. I don't use open source software because it is "morally right", or because closed source software is "evil" or full of bugs. I work as a professional embedded developer, and also as an IT and network manager, and I prefer FOSS over equivalent closed source software in many cases because it helps me do a better job. I hope that my comments here will let you see a little of my reasoning.

mvh.,

David

.